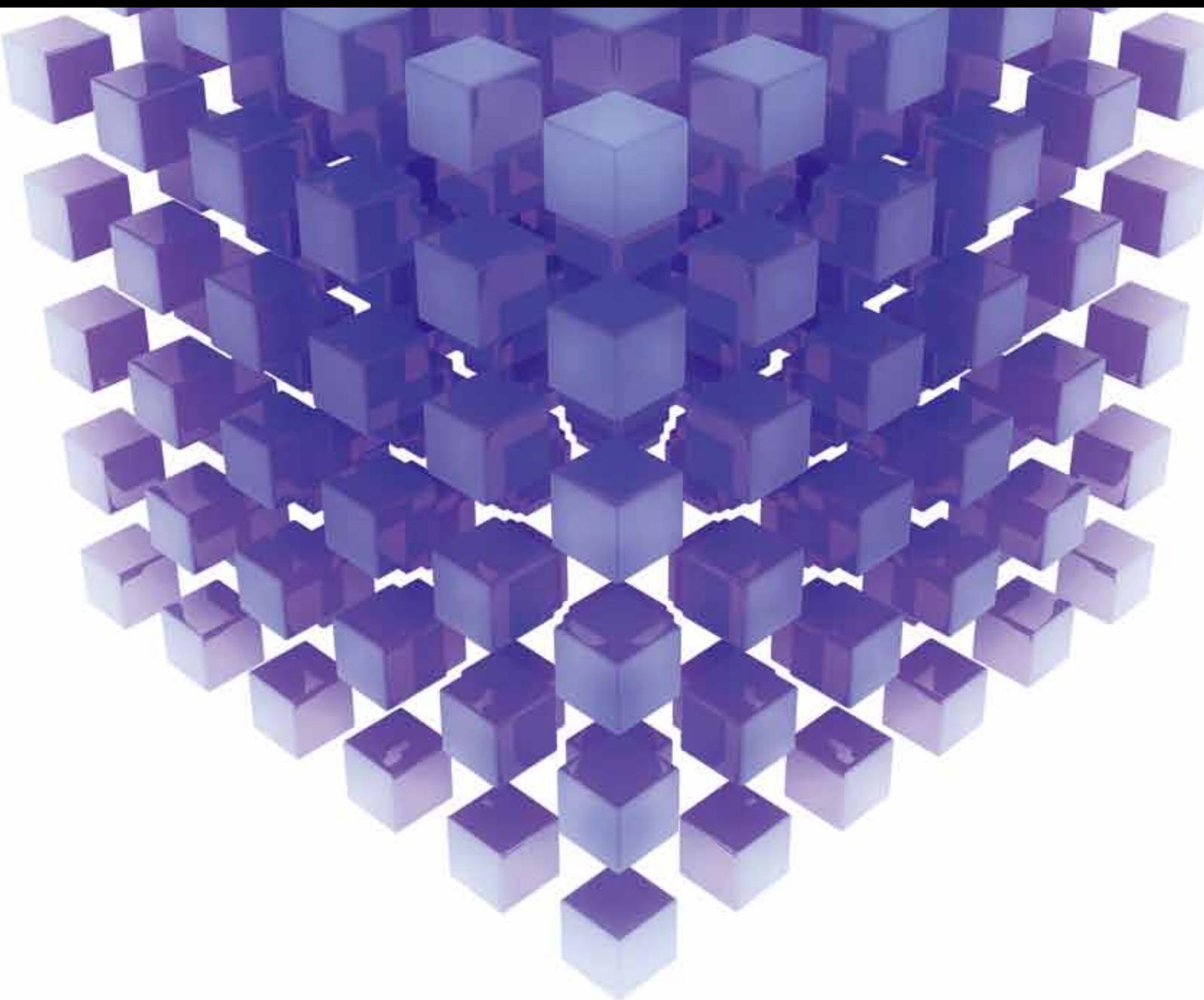


MATHEMATICAL PROBLEMS IN ENGINEERING

# ARTIFICIAL INTELLIGENCE AND ITS APPLICATIONS

GUEST EDITORS: YUDONG ZHANG, SAEED BALOCHIAN, PRAVEEN AGARWAL,  
VISHAL BHATNAGAR, AND ORWA JABER HOUSHEYA





---

# **Artificial Intelligence and Its Applications**

Mathematical Problems in Engineering

---

## **Artificial Intelligence and Its Applications**

Guest Editors: Yudong Zhang, Saeed Balochian,  
Praveen Agarwal, Vishal Bhatnagar, and Orwa Jaber Housheya



---

Copyright © 2014 Hindawi Publishing Corporation. All rights reserved.

This is a special issue published in “Mathematical Problems in Engineering.” All articles are open access articles distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

## Editorial Board

- Mohamed Abd El Aziz, Egypt  
Eihab M. Abdel-Rahman, Canada  
Rashid K. Abu Al-Rub, USA  
Sarp Adali, South Africa  
Salvatore Alfonzetti, Italy  
Igor Andrianov, Germany  
Sebastian Anita, Romania  
W. Assawinchaichote, Thailand  
Er-wei Bai, USA  
Ezzat G. Bakhoum, USA  
José Manoel Balthazar, Brazil  
Rasajit Kumar Bera, India  
Jonathan N. Blakely, USA  
Stefano Boccaletti, Spain  
Stephane P. A. Bordas, USA  
Daniela Boso, Italy  
M. Boutayeb, France  
Michael J. Brennan, UK  
Salvatore Caddemi, Italy  
Piermarco Cannarsa, Italy  
Jose E. Capilla, Spain  
Carlo Cattani, Italy  
Marcelo Cavalcanti, Brazil  
Diego J. Celentano, Chile  
Mohammed Chadli, France  
Arindam Chakraborty, USA  
Yong-Kui Chang, China  
Michael J. Chappell, UK  
Kui Fu Chen, China  
Kue-Hong Chen, Taiwan  
Xinkai Chen, Japan  
Jyh-Horng Chou, Taiwan  
Slim Choura, Tunisia  
Cesar Cruz-Hernandez, Mexico  
Erik Cuevas, Mexico  
Swagatam Das, India  
Filippo de Monte, Italy  
Yannis Dimakopoulos, Greece  
Baocang Ding, China  
Joao B. R. Do Val, Brazil  
Daoyi Dong, Australia  
B. Dubey, India  
Horst Ecker, Austria  
M. Onder Efe, Turkey  
Elmetwally Elabbasy, Egypt
- Alex Elías-Zúñiga, Mexico  
Anders Eriksson, Sweden  
Vedat S. Erturk, Turkey  
Moez Feki, Tunisia  
Ricardo Femat, Mexico  
Robertt Fontes Valente, Portugal  
Claudio Fuerte-Esquivel, Mexico  
Zoran Gajic, USA  
Ugo Galvanetto, Italy  
Xin-Lin Gao, USA  
Furong Gao, Hong Kong  
Behrouz Gatmiri, Iran  
Oleg V. Gendelman, Israel  
Paulo Batista Gonçalves, Brazil  
Oded Gottlieb, Israel  
Fabrizio Greco, Italy  
Quang Phuc Ha, Australia  
Tony Sheu Wen Hann, Taiwan  
Thomas Hanne, Switzerland  
Katica R. Hedrih, Serbia  
M. I. Herreros, Spain  
Wei-Chiang Hong, Taiwan  
Jaromir Horacek, Czech Republic  
Gordon Huang, Canada  
Huabing Huang, China  
Chuangxia Huang, China  
Yi Feng Hung, Taiwan  
Hai-Feng Huo, China  
Asier Ibeas, Spain  
Anuar Ishak, Malaysia  
Reza Jazar, Australia  
Zhijian Ji, China  
Jun Jiang, China  
J. J. Judice, Portugal  
Tadeusz Kaczorek, Poland  
Tamas Kalmar-Nagy, USA  
Tomasz Kapitaniak, Poland  
Hamid R. Karimi, Norway  
Metin O. Kaya, Turkey  
Farzad Khani, Iran  
Ren-Jieh Kuo, Taiwan  
Jurgen Kurths, Germany  
Claude Lamarque, France  
Usik Lee, Korea  
Marek Lefik, Poland
- Stefano Lenci, Italy  
Roman Lewandowski, Poland  
Shihua Li, China  
Ming Li, China  
S. Li, Canada  
Jian Li, China  
Teh-Lu Liao, Taiwan  
Panos Liatsis, UK  
Kim Meow Liew, Hong Kong  
Yi-Kuei Lin, Taiwan  
Shueei M. Lin, Taiwan  
Jui-Sheng Lin, Taiwan  
Wanquan Liu, Australia  
Bin Liu, Australia  
Yuji Liu, China  
Paolo Lonetti, Italy  
Vassilios C. Loukopoulos, Greece  
Chien-Yu Lu, Taiwan  
Junguo Lu, China  
Alexei Mailybaev, Brazil  
Manoranjan K. Maiti, India  
Oluwole Daniel Makinde, South Africa  
Rafael Martínez-Guerra, Mexico  
Driss Mehdi, France  
Roderick Melnik, Canada  
Xinzhu Meng, China  
Jose Merodio, Spain  
Yuri Vladimirovich Mikhlin, Ukraine  
Gradimir Milovanović, Serbia  
Ebrahim Momoniat, South Africa  
Trung Nguyen Thoi, Vietnam  
Hung Nguyen-Xuan, Vietnam  
Ben T. Nohara, Japan  
Sotiris K. Ntouyas, Greece  
Claudio Padra, Argentina  
Bijaya Ketan Panigrahi, India  
Francesco Pellicano, Italy  
Matjaž Perc, Slovenia  
Vu Ngoc Phat, Vietnam  
Maria do Rosário Pinho, Portugal  
Seppo Pohjolainen, Finland  
Stanislav Potapenko, Canada  
Sergio Preidikman, USA  
Carsten Proppe, Germany  
Hector Puebla, Mexico

Justo Puerto, Spain  
Dane Quinn, USA  
Kumbakonam Rajagopal, USA  
Gianluca Ranzi, Australia  
Sivaguru Ravindran, USA  
G. Rega, Italy  
Pedro Ribeiro, Portugal  
J. Rodellar, Spain  
Rosana Rodriguez-Lopez, Spain  
Alejandro J. Rodriguez-Luis, Spain  
Carla Roque, Portugal  
Rubén Ruiz García, Spain  
Manouchehr Salehi, Iran  
Miguel A. F. Sanjuán, Spain  
Ilmar Ferreira Santos, Denmark  
Nickolas S. Sapidis, Greece  
Evangelos J. Sapountzakis, Greece  
Bozidar Sarler, Slovenia  
Andrey V. Savkin, Australia  
Massimo Scalia, Italy  
Mohamed A. Seddeek, Egypt  
Leonid Shaikhet, Ukraine  
Cheng Shao, China  
Bo Shen, Germany  
Jian-Jun Shu, Singapore  
Zhan Shu, UK  
Dan Simon, USA  
Luciano Simoni, Italy

Grigori M. Sisoiev, UK  
Christos H. Skiadas, Greece  
Davide Spinello, Canada  
Sri Sridharan, USA  
Hari M. Srivastava, Canada  
Rolf Stenberg, Finland  
Changyin Sun, China  
Xi-Ming Sun, China  
Jitao Sun, China  
Andrzej Swierniak, Poland  
Yang Tang, Germany  
Allen Tannenbaum, USA  
Cristian Toma, Romania  
Gerard Olivar Tost, Colombia  
Irina N. Trendafilova, UK  
Alberto Trevisani, Italy  
Jung-Fa Tsai, Taiwan  
Kuppapalle Vajravelu, USA  
Victoria Vampa, Argentina  
Josep Vehi, Spain  
Stefano Vidoli, Italy  
Yijing Wang, China  
Cheng C. Wang, Taiwan  
Dan Wang, China  
Xiaojun Wang, China  
Qing-Wen Wang, China  
Yongqi Wang, Germany  
Moran Wang, China

Youqing Wang, China  
Gerhard-Wilhelm Weber, Turkey  
Jeroen Witteveen, The Netherlands  
Kwok-Wo Wong, Hong Kong  
Ligang Wu, China  
Zhengguang Wu, China  
Gongnan Xie, China  
Wang Xing-yuan, China  
Xi Frank Xu, China  
Xuping Xu, USA  
Jun-Juh Yan, Taiwan  
Xing-Gang Yan, UK  
Suh-Yuh Yang, Taiwan  
Mahmoud T. Yassen, Egypt  
Mohammad I. Younis, USA  
Bo Yu, China  
Huang Yuan, Germany  
S.P. Yung, Hong Kong  
Ion Zaballa, Spain  
Ashraf M. Zenkour, Saudi Arabia  
Jianming Zhan, China  
Yingwei Zhang, China  
Xu Zhang, China  
Lu Zhen, China  
Liancun Zheng, China  
Jian Guo Zhou, UK  
Zexuan Zhu, China  
Mustapha Zidi, France

# Contents

**Artificial Intelligence and Its Applications**, Yudong Zhang, Saeed Balochian, Praveen Agarwal, Vishal Bhatnagar, and Orwa Jaber Housheya  
Volume 2014, Article ID 840491, 10 pages

**The Application of FastICA Combined with Related Function in Blind Signal Separation**, Dengao Li, Junmin Zhao, Hongyan Liu, and Defeng Hao  
Volume 2014, Article ID 953745, 9 pages

**Towards a Unified Sentiment Lexicon Based on Graphics Processing Units**, Liliana Ibeth Barbosa-Santillán and Inmaculada Álvarez-de-Mon y-Rego  
Volume 2014, Article ID 429629, 19 pages

**Wolf Pack Algorithm for Unconstrained Global Optimization**, Hu-Sheng Wu and Feng-Ming Zhang  
Volume 2014, Article ID 465082, 17 pages

**Research on Cooperative Combat for Integrated Reconnaissance-Attack-BDA of Group LAVs**, Li Bing, Li Jie, He Guanglin, and Li Dalin  
Volume 2014, Article ID 123142, 6 pages

**Optimum Performance-Based Seismic Design Using a Hybrid Optimization Algorithm**, S. Talatahari, A. Hosseini, S. R. Mirghaderi, and F. Rezazadeh  
Volume 2014, Article ID 693128, 8 pages

**Incremental Tensor Principal Component Analysis for Handwritten Digit Recognition**, Chang Liu, Tao Yan, WeiDong Zhao, YongHong Liu, Dan Li, Feng Lin, and JiLiu Zhou  
Volume 2014, Article ID 819758, 10 pages

**Structural Reliability Assessment by Integrating Sensitivity Analysis and Support Vector Machine**, Shao-Fei Jiang, Da-Bao Fu, and Si-Yao Wu  
Volume 2014, Article ID 586191, 6 pages

**An Algorithm for Mining of Association Rules for the Information Communication Network Alarms Based on Swarm Intelligence**, Yang Wang, Guocai Li, Yakun Xu, and Jie Hu  
Volume 2014, Article ID 894205, 14 pages

**Hybrid Functional-Neural Approach for Surface Reconstruction**, Andrés Iglesias and Akemi Gálvez  
Volume 2014, Article ID 351648, 13 pages

**Solving the Balanced Academic Curriculum Problem Using the ACO Metaheuristic**, José-Miguel Rubio, Wenceslao Palma, Nibaldo Rodriguez, Ricardo Soto, Broderick Crawford, Fernando Paredes, and Guillermo Cabrera  
Volume 2013, Article ID 793671, 8 pages

**A Genetic-Algorithms-Based Approach for Programming Linear and Quadratic Optimization Problems with Uncertainty**, Weihua Jin, Zhiying Hu, and Christine W. Chan  
Volume 2013, Article ID 272491, 12 pages

**Composite Broadcasting and Ranging via a Satellite Dual-Frequency MPPSK System**, Yu Yao, Lenan Wu, and Jiwu Wang  
Volume 2013, Article ID 404357, 8 pages

**Research on ISFLA-Based Optimal Control Strategy for the Coordinated Charging of EV Battery Swap Station**, Xueliang Huang, Hao Qiang, Qidong Zhang, and Haijuan Li  
Volume 2013, Article ID 613404, 7 pages

**The Study of Reinforcement Learning for Traffic Self-Adaptive Control under Multiagent Markov Game Environment**, Lun-Hui Xu, Xin-Hai Xia, and Qiang Luo  
Volume 2013, Article ID 962869, 10 pages

**Application of Adaptive Extended Kalman Smoothing on INS/WSN Integration System for Mobile Robot Indoors**, Xiyuan Chen, Yuan Xu, and Qinghua Li  
Volume 2013, Article ID 130508, 8 pages

**Entropy-Based and Weighted Selective SIFT Clustering as an Energy Aware Framework for Supervised Visual Recognition of Man-Made Structures**, Ayman El Mobacher, Nicholas Mitri, and Mariette Awad  
Volume 2013, Article ID 730143, 9 pages

**Hyperspectral Image Classification Using Kernel Fukunaga-Koontz Transform**, Semih Dinç and Abdullah Bal  
Volume 2013, Article ID 471915, 7 pages

**Solving a Novel Inventory Location Model with Stochastic Constraints and  $(R, s, S)$  Inventory Control Policy**, Guillermo Cabrera, Pablo A. Miranda, Enrique Cabrera, Ricardo Soto, Broderick Crawford, Jose Miguel Rubio, and Fernando Paredes  
Volume 2013, Article ID 670528, 12 pages

**Genetic Pattern Search and Its Application to Brain Image Classification**, Yudong Zhang, Shuihua Wang, Genlin Ji, and Zhengchao Dong  
Volume 2013, Article ID 580876, 8 pages

**Modeling and Analysis of the Weld Bead Geometry in Submerged Arc Welding by Using Adaptive Neurofuzzy Inference System**, Nuri Akkas, Durmuş Karayel, Sinan Serdar Ozkan, Ahmet Oğur, and Bayram Topal  
Volume 2013, Article ID 473495, 10 pages

**Nonlinear Predictive Control of Mass Moment Aerospace Vehicles Based on Ant Colony Genetic Algorithm Optimization**, Xiaoyu Zhang, Peng Li, Dexin Xu, Ben Mao, and Kunpeng He  
Volume 2013, Article ID 858190, 10 pages

**A Multilayer Hidden Markov Models-Based Method for Human-Robot Interaction**, Chongben Tao and Guodong Liu  
Volume 2013, Article ID 384865, 12 pages

**Matching Cost Filtering for Dense Stereo Correspondence**, Yimin Lin, Naiguang Lu, Xiaoping Lou, Fang Zou, Yanbin Yao, and Zhaocai Du  
Volume 2013, Article ID 654139, 11 pages

**Knowledge Mining Based on Environmental Simulation Applied to Wind Farm Power Forecasting**, Dongxiao Niu, Ling Ji, Qingguo Ma, and Wei Li  
Volume 2013, Article ID 597562, 8 pages

**Identification of Code-Switched Sentences and Words Using Language Modeling Approaches,**  
Liang-Chih Yu, Wei-Cheng He, Wei-Nan Chien, and Yuen-Hsien Tseng  
Volume 2013, Article ID 898714, 7 pages

**Extraction of Belief Knowledge from a Relational Database for Quantitative Bayesian Network Inference,** LiMin Wang  
Volume 2013, Article ID 297121, 10 pages

**Robust Quadratic Regression and Its Application to Energy-Growth Consumption Problem,**  
Yongzhi Wang, Yuli Zhang, Fuliang Zhang, and Jining Yi  
Volume 2013, Article ID 210510, 10 pages

**LGMS-FOA: An Improved Fruit Fly Optimization Algorithm for Solving Optimization Problems,**  
Dan Shan, GuoHua Cao, and HongJiang Dong  
Volume 2013, Article ID 108768, 9 pages

**Improved SpikeProp for Using Particle Swarm Optimization,** Falah Y. H. Ahmed,  
Siti Mariyam Shamsuddin, and Siti Zaiton Mohd Hashim  
Volume 2013, Article ID 257085, 13 pages

**Nighttime Fire/Smoke Detection System Based on a Support Vector Machine,** Chao-Ching Ho  
Volume 2013, Article ID 428545, 7 pages

**Efficient Secure Multiparty Computation Protocol for Sequencing Problem over Insecure Channel,**  
Yi Sun, Qiaoyan Wen, Yudong Zhang, Hua Zhang, and Zhengping Jin  
Volume 2013, Article ID 172718, 5 pages

**Study on Semi-Parametric Statistical Model of Safety Monitoring of Cracks in Concrete Dams,**  
Chongshi Gu, Dong Qin, Zhanchao Li, and Xueqin Zheng  
Volume 2013, Article ID 874629, 9 pages

**A Hybrid Bat Algorithm with Path Relinking for Capacitated Vehicle Routing Problem,** Yongquan Zhou,  
Jian Xie, and Hongqing Zheng  
Volume 2013, Article ID 392789, 10 pages

**Ripple-Spreading Network Model Optimization by Genetic Algorithm,** Xiao-Bing Hu, Ming Wang,  
and Mark S. Leeson  
Volume 2013, Article ID 176206, 15 pages

**Tundish Cover Flux Thickness Measurement Method and Instrumentation Based on Computer Vision in Continuous Casting Tundish,** Meng Lu, He Qing, Xie Zhi, Yang Weimin, Ci Ying, Zhang Chuanyi,  
and Gao Hongliang  
Volume 2013, Article ID 690948, 15 pages

**Context Prediction of Mobile Users Based on Time-Inferred Pattern Networks: A Probabilistic Approach,** Yong-Hyuk Kim and Yourim Yoon  
Volume 2013, Article ID 106139, 10 pages

**Emotion Expression of Robot with Personality,** Xue Hu, Lun Xie, Xin Liu, and Zhiliang Wang  
Volume 2013, Article ID 132735, 10 pages

**Multiagent Reinforcement Learning with Regret Matching for Robot Soccer**, Qiang Liu, Jiachen Ma, and Wei Xie  
Volume 2013, Article ID 926267, 8 pages

**A Wavelet Kernel-Based Primal Twin Support Vector Machine for Economic Development Prediction**, Fang Su and HaiYang Shang  
Volume 2013, Article ID 875392, 6 pages

**Optimal Design of Signal Controlled Road Networks Using Differential Evolution Optimization Algorithm**, Huseyin Ceylan  
Volume 2013, Article ID 696374, 11 pages

**Interesting Activities Discovery for Moving Objects Based on Collaborative Filtering**, Guan Yuan, Shixiong Xia, and Yanmei Zhang  
Volume 2013, Article ID 380871, 9 pages

**Surface Defect Target Identification on Copper Strip Based on Adaptive Genetic Algorithm and Feature Saliency**, Xuewu Zhang, Wei Li, Ji Xi, Zhuo Zhang, and Xinnan Fan  
Volume 2013, Article ID 504895, 10 pages

**Research on the Production Scheduling Optimization for Virtual Enterprises**, Min Huang, Ruixian Huang, Bo Sun, and Linrong Li  
Volume 2013, Article ID 492158, 9 pages

**Semisupervised Clustering for Networks Based on Fast Affinity Propagation**, Mu Zhu, Fanrong Meng, and Yong Zhou  
Volume 2013, Article ID 385265, 13 pages

**UCAV Path Planning by Fitness-Scaling Adaptive Chaotic Particle Swarm Optimization**, Yudong Zhang, Lenan Wu, and Shuihua Wang  
Volume 2013, Article ID 705238, 9 pages

**CFSO<sup>3</sup>: A New Supervised Swarm-Based Optimization Algorithm**, Antonino Laudani, Francesco Riganti Fulginei, Alessandro Salvini, Maurizio Schmid, and Silvia Conforto  
Volume 2013, Article ID 560614, 13 pages

**Weighted-Bit-Flipping-Based Sequential Scheduling Decoding Algorithms for LDPC Codes**, Qing Zhu and Le-nan Wu  
Volume 2013, Article ID 371206, 6 pages

**Artificial Bee Colony Algorithm Merged with Pheromone Communication Mechanism for the 0-1 Multidimensional Knapsack Problem**, Junzhong Ji, Hongkai Wei, Chunnian Liu, and Baocai Yin  
Volume 2013, Article ID 676275, 13 pages

**Visual Object Tracking Based on 2DPCA and ML**, Ming-Xin Jiang, Min Li, and Hong-Yu Wang  
Volume 2013, Article ID 404978, 7 pages

**Practical Speech Emotion Recognition Based on Online Learning: From Acted Data to Elicited Data**, Chengwei Huang, Ruiyu Liang, Qingyun Wang, Ji Xi, Cheng Zha, and Li Zhao  
Volume 2013, Article ID 265819, 9 pages

**A Wavelet-Based Robust Relevance Vector Machine Based on Sensor Data Scheduling Control for Modeling Mine Gas Gushing Forecasting on Virtual Environment**, Wang Ting, Cai Lin-qin, Fu Yao, and Zhu Tingcheng

Volume 2013, Article ID 579693, 4 pages

**Neural Model with Particle Swarm Optimization Kalman Learning for Forecasting in Smart Grids**, Alma Y. Alanis, Luis J. Ricalde, Chiara Simetti, and Francesca Odone

Volume 2013, Article ID 197690, 9 pages

**A Novel Method for Surface Defect Detection of Photovoltaic Module Based on Independent Component Analysis**, Xuewu Zhang, Hao Sun, Yun Zhou, Ji Xi, and Min Li

Volume 2013, Article ID 520568, 8 pages

**Weight-Selected Attribute Bagging for Credit Scoring**, Jianwu Li, Haizhou Wei, and Wangli Hao

Volume 2013, Article ID 379690, 13 pages

**Design of Special Impacting Filter for Multicarrier ABPSK System**, Zhimin Chen and Lenan Wu

Volume 2013, Article ID 921932, 7 pages

**Fault Diagnosis for Wireless Sensor by Twin Support Vector Machine**, Mingli Ding, Dongmei Yang, and Xiaobing Li

Volume 2013, Article ID 718783, 5 pages

**Solving Two-Dimensional HP Model by Firefly Algorithm and Simplified Energy Function**,

Yudong Zhang, Lenan Wu, and Shuihua Wang

Volume 2013, Article ID 398141, 9 pages

## Editorial

# Artificial Intelligence and Its Applications

**Yudong Zhang,<sup>1</sup> Saeed Balochian,<sup>2</sup> Praveen Agarwal,<sup>3</sup>  
Vishal Bhatnagar,<sup>4</sup> and Orwa Jaber Housheya<sup>5</sup>**

<sup>1</sup> School of Computer Science and Technology, Nanjing Normal University, Nanjing, Jiangsu 210023, China

<sup>2</sup> Department of Electrical Engineering, Gonabad Branch, Islamic Azad University, Gonabad, Khorasan-e-Razavi 96916-29, Iran

<sup>3</sup> Department of Mathematics, Anand International College of Engineering, Agra Road, Near Bassi, Jaipur, Rajasthan 303012, India

<sup>4</sup> Ambedkar Institute of Advanced Communication Technologies and Research, Government of NCT of Delhi, Geeta Colony, Delhi 110031, India

<sup>5</sup> Department of Chemistry, Arab American University, 19356 Jenin, Palestine

Correspondence should be addressed to Yudong Zhang; zhangyudongnuaa@gmail.com

Received 18 March 2014; Accepted 18 March 2014; Published 10 April 2014

Copyright © 2014 Yudong Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Artificial intelligence (AI) has revolutionized information technology. The new economy of information technology has shaped the way we are living. Recently, AI algorithms have attracted close attention of researchers and have also been applied successfully to solve problems in engineering. Nevertheless, for large and complex problems, AI algorithms consume considerable computation time due to stochastic feature of the search approaches. Therefore, there is a potential requirement to develop efficient algorithm to find solutions under the limited resources, time, and money in real-world applications. This special issue aims to report the latest advances in every aspect of artificial intelligence technology, including machine learning, data mining, computer vision, multiagent systems, evolutionary computation, and fuzzy logic.

The special issue received 142 high quality submissions from different countries all over the world. All submitted papers followed the same standard (peer-reviewed by at least three independent reviewers) as applied to regular submissions to *Mathematical Problems in Engineering*. Due to the limited space, 57 papers were finally included. The primary guideline was to demonstrate the most significant developments on the topics of AI and to apply AI algorithms in real-life scenarios.

In the paper entitled “*Solving two-dimensional HP model by firefly algorithm and simplified energy function*,” Y. Zhang et al. investigate traditional energy function and point out

that its discrete property cannot give direction of the next step to the searching point, causing a challenge to optimization algorithms. Therefore, they introduce the simplified energy function to turn traditional discrete energy function to a continuous one. The simplified energy function totals the distance between all pairs of hydrophobic amino acids. To optimize the simplified energy function, they introduce the latest swarm intelligence algorithm, the firefly algorithm (FA). The experiments take 14 sequences of different chain lengths from 18 to 100 as the dataset and compare the FA with standard genetic algorithm and immune genetic algorithm. Each algorithm runs 20 times. The averaged energy convergence results show that FA achieves the lowest values. It concludes that it is effective to solve 2D HP model by the FA and the simplified energy function.

In the paper entitled “*Fault diagnosis for wireless sensor by twin support vector machine*,” M. Ding et al. propose a novel fault diagnosis method for wireless sensor technology by twin support vector machine (TSVM) in order to improve the diagnosis accuracy of wireless sensor. Twin SVM is a binary classifier that performs classification by using two nonparallel hyperplanes instead of the single hyperplane used in the classical SVM. However, the parameter setting in the TSVM training procedure significantly influences the classification accuracy. Their study introduces PSO as an optimization technique to simultaneously optimize the TSVM training parameter. The experimental results indicate

that the diagnosis results for wireless sensor of twin support vector machine are better than those of SVM, ANN.

In the paper “*Design of special impacting filter for multicarrier ABPSK system*,” Z. Chen and L. Wu propose an iterative scheme to derive the parameters of the cascade notch filter. The cost function is determined by the cascading notch filter’s influence on impacting filters, converting the cost function’s least square problem to a filter parameters’ standard quadratic programming problem. Finally, a cascading notch SIF (CNSIF) designed to demodulate the ABPSK signals is realized.

In the paper “*Weight-selected attribute bagging for credit scoring*,” J. Li et al. propose an improved attribute bagging method, weight-selected attribute bagging (WSAB), to evaluate credit risk. Weights of attributes are first computed using attribute evaluation method such as linear support vector machine (LSVM) and principal component analysis (PCA). Subsets of attributes are then constructed according to weights of attributes. For each of attribute subsets, the larger the weights of the attributes, the larger the probabilities by which they are selected into the attribute subset. Next, training samples and test samples are projected onto each attribute subset, respectively. A scoring model is then constructed based on each set of newly produced training samples. Finally, all scoring models are used to vote for test instances. An individual model that only uses selected attributes will be more accurate because of elimination of some of redundant and uninformative attributes. Besides, the way of selecting attributes by probability can also guarantee the diversity of scoring models. Experimental results based on two credit benchmark databases show that the proposed method, WSAB, is outstanding in both prediction accuracy and stability, as compared to analogous methods.

In the paper entitled “*A novel method for surface defect detection of photovoltaic module based on independent component analysis*,” the research of X. Zhang et al. proposes a new method for surface defect detection of photovoltaic module based on independent component analysis (ICA) reconstruction algorithm. Firstly, a faultless image is used as the training image. The demixing matrix and corresponding ICs are obtained by applying the ICA in the training image. Then they reorder the ICs according to the range values and reform the demixing matrix. The reformed demixing matrix is used to reconstruct the defect image. The resulting image can remove the background structures and enhance the local anomalies. Experimental results show that the proposed method can effectively detect the presence of defects in periodically patterned surfaces.

In the paper entitled “*Neural model with particle swarm optimization Kalman learning for forecasting in smart grids*,” A. Y. Alanis et al. discuss a novel training algorithm for a neural network architecture applied to time series prediction with smart grids applications. The proposed training algorithm is based on an extended Kalman filter (EKF) improved using particle swarm optimization (PSO) to compute the design parameters. The EKF-PSO-based algorithm is employed to update the synaptic weights of the neural network. The size of the regression vector is determined by means of the Cao methodology. The proposed structure captures more

efficiently the complex nature of the wind speed, energy generation, and electrical load demand time series that are constantly monitored in a smart grid benchmark. The proposed model is trained and tested using real data values in order to show the applicability of the proposed scheme.

In the paper entitled “*A wavelet-based robust relevance vector machine based on sensor data scheduling control for modeling mine gas gushing forecasting on virtual environment*,” W. Ting et al. present a wavelet-based robust relevance vector machine based on sensor data scheduling control for modeling mine gas gushing forecasting. Morlet wavelet function can be used as the kernel function of robust relevance vector machine. Mean percentage error has been used to measure the performance of the proposed method in this study. As the mean prediction error of mine gas gushing of the WRRVM model is less than 1.5% and the mean prediction error of mine gas gushing of the RVM model is more than 2.5%, it can be seen that the prediction accuracy for mine gas gushing of the WRRVM model is better than that of the RVM model.

In the paper entitled “*Practical speech emotion recognition based on online learning: from acted data to elicited data*,” C. Huang et al. study the cross-database speech emotion recognition based on online learning. How to apply a classifier trained on acted data to naturalistic data, such as elicited data, remains a major challenge in today’s speech emotion recognition system. They introduce three types of different data sources: first, a basic speech emotion dataset collected from acted speech by professional actors and actresses; second, a speaker-independent data set which contains a large number of speakers; third, an elicited speech data set collected from a cognitive task. Acoustic features are extracted from emotional utterances and evaluated by using maximal information coefficient (MIC). A baseline valence and arousal classifier is designed based on Gaussian mixture models. Online training module is implemented by using AdaBoost. While the offline recognizer is trained on the acted data, the online testing data includes the speaker-independent data and the elicited data. Experimental results show that by introducing the online learning module their speech emotion recognition system can be better adapted to new data, which is an important character in real world applications.

In the paper entitled “*Visual object tracking based on 2DPCA and ML*,” M.-X. Jiang et al. present a novel visual object tracking algorithm based on two-dimensional principal component analysis (2DPCA) and maximum likelihood estimation (MLE). Firstly, they introduce regularization into the 2DPCA reconstruction and develop an iterative algorithm to represent an object by 2DPCA bases. Secondly, the model of sparsity constrained MLE is established. Abnormal pixels in the samples will be assigned with low weights to reduce the effects on the tracking algorithm. The object tracking results are obtained by using Bayesian maximum a posteriori (MAP) probability estimation. Finally, to further reduce tracking drift, they employ a template update strategy that combines incremental subspace learning and the error matrix. Their strategy adapts the template to the appearance change of the target and reduces the influence of the occluded target template as well. Compared with other popular methods, their

method reduces the computational complexity and is very robust to abnormal changes. Both qualitative and quantitative evaluations on challenging image sequences demonstrate that the proposed tracking algorithm achieves more favorable performance than several state-of-the-art methods.

In the paper entitled “*Artificial bee colony algorithm merged with pheromone communication mechanism for the 0-1 multidimensional knapsack problem*,” J. Ji et al. propose a new artificial bee colony (ABC) algorithm for the 0-1 multidimensional knapsack problem (MKP\_01). The new ABC algorithm introduces a novel communication mechanism among bees, which bases on the updating and diffusion of inductive pheromone produced by bees. In a number of experiments and comparisons, their approach obtains better quality solutions in shorter time than the ABC algorithm without the mechanism. They also compare the solution performance of the proposed approach against some stochastic approaches recently reported in the literature. Computational results demonstrate the superiority of the new ABC approach over all the other approaches.

In another paper, Q. Zhu and L.-N. Wu present “*Weighted-bit-flipping-based sequential scheduling decoding algorithms for LDPC codes*.” Their paper proposes a sequential scheduling algorithm based on weighted bit-flipping (WBF) algorithm for the sake of improving the convergence speed. Notoriously, WBF is a low-complexity and simple algorithm. They combine it with belief propagation (BP) to obtain advantages of these two algorithms. Flipping function used in WBF is borrowed to determine the priority of scheduling. Simulation results show that it can provide a good tradeoff between FER performance and computation complexity for short-length low-density parity-check (LDPC) codes.

In another paper, the research of A. Laudani et al. entitled “*CFSO<sup>3</sup>: a new supervised swarm-based optimization algorithm*,” presents CFSO<sup>3</sup>, an optimization heuristic within the class of the swarm intelligence, based on a synergy among three different features of the Continuous flock-of-starlings optimization. One of the main novelties is that this optimizer is no more a classical numerical algorithm since it now can be seen as a continuous dynamic system, which can be treated by using all the mathematical instruments available for managing state equations. In addition, CFSO<sup>3</sup> allows passing from stochastic approaches to supervised deterministic ones since the random updating of parameters, a typical feature for numerical swarm-based optimization algorithms, is now fully substituted by a supervised strategy: in CFSO<sup>3</sup> the tuning of parameters is a priori designed for obtaining both exploration and exploitation. Indeed the exploration, that is, the escaping from a local minimum, as well as the convergence and the refinement to a solution can be designed simply by managing the eigenvalues of the CFSO state equations. Virtually in CFSO<sup>3</sup>, just the initial values of positions and velocities of the swarm members have to be randomly assigned. Both standard and parallel versions of CFSO<sup>3</sup> together with validations on classical benchmarks are presented.

In the paper “*UCAV path planning by fitness-scaling adaptive chaotic particle swarm optimization*,” Y. Zhang et

al. propose a fitness-scaling adaptive chaotic particle swarm optimization (FAC-PSO) approach as a fast and robust approach for the task of path planning of UCAVs. The FAC-PSO employed the fitness-scaling method, the adaptive parameter mechanism, and the chaotic theory. Experiments show that the FAC-PSO is more robust and costs less time than elite genetic algorithm with migration, simulated annealing, and chaotic artificial bee colony. Moreover, the FAC-PSO performs well on the application of dynamic path planning when the threats cruise randomly and on the application of 3D path planning.

In the paper entitled “*Semisupervised clustering for networks based on fast affinity propagation*,” M. Zhu et al. propose a semisupervised clustering algorithm for networks based on fast affinity propagation (SCAN-FAP), which is essentially a kind of similarity metric learning method. Firstly, they define a new constraint similarity measure integrating the structural information and the pairwise constraints, which reflects the effective similarities between nodes in networks. Then, taking the constraint similarities as input, they propose a fast affinity propagation algorithm that keeps the advantages of the original affinity propagation algorithm while increasing the time efficiency by passing only the messages between certain nodes. Finally, by extensive experimental studies, they demonstrate that the proposed algorithm can take full advantage of the prior knowledge and improve the clustering quality significantly. Furthermore, the proposed algorithm has a superior performance to some of the state-of-the-art approaches.

In the paper entitled “*Research on the production scheduling optimization for virtual enterprises*,” M. Huang et al. establish a partner selection model based on an improved ant colony algorithm, then present a production scheduling framework with two layers as global scheduling and local scheduling for virtual enterprise, and give a global scheduling mathematical model with the smallest total production time based on it. An improved genetic algorithm is proposed in the model to solve the time complexity of virtual enterprise production scheduling. The experimental results validate the optimization of the model and the efficiency of the algorithm.

In the paper entitled “*Surface defect target identification on copper strip based on adaptive genetic algorithm and feature saliency*,” X. Zhang et al. propose a new surface defect target identification method for copper strip based on adaptive genetic algorithm (AGA) and feature saliency. First, the study uses gray level cooccurrence matrix (GLCM) and HU invariant moments for feature extraction. Then, adaptive genetic algorithm, which is used for feature selection, is evaluated and discussed. In AGA, total error rates and false alarm rates are integrated to calculate the fitness value, and the probability of crossover and mutation is adjusted dynamically according to the fitness value. At last, the selected features are optimized in accordance with feature saliency and are inputted into a support vector machine (SVM). Furthermore, for comparison, they conduct experiments using the selected optimal feature subsequence (OFS) and the total feature sequence (TFS) separately. The experimental results demonstrate that the proposed method can guarantee

the correct rates of classification and can lower the false alarm rates.

In the paper entitled “*Interesting activities discovery for moving objects based on collaborative filtering*,” G. Yuan et al. propose a method of interesting activities discovery based on collaborative filtering. First, the interesting degree of the objects’ activities is calculated comprehensively. Then, combined with the newly proposed hybrid collaborative filtering, similar objects can be computed and all kinds of interesting activities can be discovered. Finally, potential activities are recommended according to their similar objects. The experimental results show that the method is effective and efficient in finding objects’ interesting activities.

In the paper entitled “*Optimal design of signal controlled road networks using differential evolution optimization algorithm*,” H. Ceylan proposes a traffic congestion minimization model in which the traffic signal setting optimization is performed through a combined simulation-optimization model. In this model, the TRANSYT traffic simulation software is combined with differential evolution (DE) optimization algorithm, which is based on the natural selection paradigm. In this context, the equilibrium network design (EQND) problem is formulated as a bilevel programming problem in which the upper level is the minimization of the total network performance index. In the lower level, the traffic assignment problem, which represents the route choice behavior of the road users, is solved using the path flow estimator (PFE) as a stochastic user equilibrium assessment. The solution of the bilevel EQND problem is carried out by the proposed differential evolution and TRANSYT with PFE, the so-called DETRANSPFE model, on a well-known signal controlled test network. Performance of the proposed model is compared to that of two previous works where the EQND problem has been solved by genetic-algorithms- (GAs-) and harmony-search- (HS-) based models. Results show that the DETRANSPFE model outperforms the GA- and HS-based models in terms of the network performance index and the computational time required.

In the paper entitled “*A wavelet kernel-based primal twin support vector machine for economic development prediction*,” F. Su and H. Shang propose an economic development prediction method based on the wavelet kernel-based primal twin support vector machine algorithm. As gross domestic product (GDP) is an important indicator to measure economic development, economic development prediction means GDP prediction in this study. The wavelet kernel-based primal twin support vector machine algorithm can solve two smaller sized quadratic programming problems instead of solving a large one as in the traditional support vector machine algorithm. Economic development data of Anhui province from 1992 to 2009 are used to study the prediction performance of the wavelet kernel-based primal twin support vector machine algorithm. The comparison of mean error of economic development prediction between wavelet kernel-based primal twin support vector machine and traditional support vector machine models trained by the training samples with the 3–5-dimensional input vectors, respectively, is given in this paper. The testing results show that the economic development prediction accuracy of the

wavelet kernel-based primal twin support vector machine model is better than that of traditional support vector machine.

In the paper entitled “*Multiagent reinforcement learning with regret matching for robot soccer*,” Q. Liu et al. propose a novel multiagent reinforcement learning (MARL) algorithm Nash-Q learning with regret matching, in which regret matching is used to speed up the well-known MARL algorithm Nash-Q learning. It is of importance to choose the suitable strategy of action selection to balance exploration and exploitation, with the aim of enhancing the ability of online learning of Nash-Q learning. In Markov game the joint action of agents adopting regret matching algorithm can converge to a group of points of no-regret that can be viewed as coarse correlated equilibrium which includes Nash equilibrium in essence. It is can be inferred that regret matching can guide exploration of the state-action space so that the rate of convergence of Nash-Q learning algorithm can be increased. Simulation results on robot soccer validate that compared to original Nash-Q learning algorithm, the use of regret matching during the learning phase of Nash-Q learning has excellent ability of online learning and results in significant performance in terms of scores, average reward, and policy convergence.

In another paper, X. Hu et al. present “*Emotion expression of robot with personality*.” They build a robot emotional expression model based on hidden Markov model (HMM) to enable robots that have different personalities to respond in a more satisfactory emotional level. Gross emotion regulation theory and five factors model (FFM) that are the theoretical basis are firstly described. The importance of the personality effect on the emotion expression process is proposed, and how to make the effect quantization is discussed. After that, the algorithm of HMM is used to describe the process of emotional state transition and expression, and the performance transferring probability affected by personality is calculated. At last, the algorithm model is simulated and applied in a robot platform. The results prove that the emotional expression model can acquire human-like expressions and improve the human-computer interaction.

In another paper, the research of Y.-H. Kim and Y. Yoon entitled “*Context prediction of mobile users based on time-inferred pattern networks: a probabilistic approach*” presents a probabilistic method of predicting context of mobile users based on their historic context data. The presented method predicts general context based on probability theory through a novel graphical data structure, which is a kind of weighted directed multigraphs. User context data are transformed into the new graphical structure, in which each node represents a context or a combined context and each directed edge indicates a context transfer with the time weight inferred from corresponding time data. They also consider the periodic property of context data and devise a good solution to context data with such property. Test shows the merits of the presented method.

In the paper “*Tundish cover flux thickness measurement method and instrumentation based on computer vision in continuous casting tundish*,” M. Lu et al. specifically design and build instrumentation and present a novel method to

measure the tundish cover flux (TCF) thickness. The instrumentation is composed of a measurement bar, a mechanical device, a high-definition industrial camera, a Siemens S7-200 programmable logic controller (PLC), and a computer. Their measurement method is based on the computer vision algorithms, including image denoising method, monocular range measurement method, scale invariant feature transform (SIFT), and image gray gradient detection method. Using the present instrumentation and method, images in the continuous casting (CC) tundish can be collected by camera and transferred to computer to do image processing. Experiments show that the proposed instrumentation and method worked well at scene of steel plants, can accurately measure the thickness of TCF, and overcome the disadvantages of traditional measurement methods, or even replace the traditional ones.

In the paper entitled "*Ripple-spreading network model optimization by genetic algorithm*," X.-B. Hu et al. attempt to apply genetic algorithm (GA) to tune the values of ripple-spreading related parameters (RSRPs), so that the ripple-spreading network model (RSNM) may generate these two most important network topologies. The study demonstrates that, once RSRPs are properly tuned by GA, the RSNM is capable of generating both network topologies and therefore has a great flexibility to study many real-world complex network systems.

In the paper entitled "*A hybrid bat algorithm with path relinking for capacitated vehicle routing problem*," Y. Zhou et al. propose a hybrid bat algorithm with path relinking (HBA-PR) to solve capacitated vehicle routing problem (CVRP). The HBA-PR is constructed based on the framework of continuous bat algorithm; the greedy randomized adaptive search procedure (GRASP) and path relinking are effectively integrated into bat algorithm. Moreover, in order to further improve the performance, the random subsequences and single-point local search are operated with certain loudness (probability). In order to verify the validity of their method, several classical CVRP instances from three classes of benchmarks are selected to test its efficiency compared with other existing methods. Experimental results and comparisons show that the HBA-PR is effective for CVRP.

In the paper entitled "*Study on semi-parametric statistical model of safety monitoring of cracks in concrete dams*," C. Gu et al. consider that cracks are one of the hidden dangers in concrete dams. The study on safety monitoring models of concrete dam cracks has always been difficult. Using the parametric statistical model of safety monitoring of cracks in concrete dams, with the help of the semiparametric statistical theory, and considering the abnormal behaviors of these cracks, the semiparametric statistical model of safety monitoring of concrete dam cracks is established to overcome the limitation of the parametric model in expressing the objective model. Previous projects show that the semiparametric statistical model has a stronger fitting effect and has a better explanation for cracks in concrete dams than the parametric statistical model. However, when used for forecast, the forecast capability of the semiparametric statistical model is equivalent to that of the parametric statistical model. The modeling of the semiparametric statistical model is simple,

has a reasonable principle, and has a strong practicality, with a good application prospect in the actual project.

In the paper entitled "*Efficient secure multiparty computation protocol for sequencing problem over insecure channel*," Y. Sun et al. believe that secure multiparty computation is more and more popular in electronic bidding, anonymous voting, and online auction, as a powerful tool in solving privacy preserving cooperative problems. Privacy preserving sequencing problem that is an essential link is regarded as the core issue in these applications. However, due to the difficulties of solving multiparty privacy preserving sequencing problem, related secure protocol is extremely rare. In order to break this deadlock, their paper presents an efficient secure multiparty computation protocol for the general privacy-preserving sequencing problem based on symmetric homomorphic encryption. The result is of value not only in theory, but also in practice.

In the paper entitled "*Nighttime fire/smoke detection system based on a support vector machine*," C.-C. Ho states that a laser light can be projected into the monitored field of view and the returning projected light section image can be analyzed to detect fire and/or smoke, in order to overcome the nighttime limitations of video smoke detection methods. If smoke appears within the monitoring zone created from the diffusion or scattering of light in the projected path, the camera sensor receives a corresponding signal. The successive processing steps of the proposed real-time algorithm use the spectral, diffusing, and scattering characteristics of the smoke-filled regions in the image sequences to register the position of possible smoke in a video. Characterization of smoke is carried out by a nonlinear classification method using a support vector machine, and this is applied to identify the potential fire/smoke location. Experimental results in a variety of nighttime conditions demonstrate that the proposed fire/smoke detection method can successfully and reliably detect fires by identifying the location of smoke.

In the paper entitled "*Improved SpikeProp for using particle swarm optimization*," F. Y. H. Ahmed et al. derive a novel supervised learning rule for SpikeProp to overcome the discontinuities introduced by the spiking thresholding. Their algorithm is based on an error-backpropagation learning rule suited for supervised learning of spiking neurons that use exact spike time coding. The SpikeProp is able to demonstrate the spiking neurons that can perform complex nonlinear classification in fast temporal coding. Their study proposes enhancements of SpikeProp learning algorithm for supervised training of spiking networks which can deal with complex patterns. The proposed methods include the SpikeProp particle swarm optimization (PSO) and angle driven dependency learning rate. These methods are presented to SpikeProp network for multilayer learning enhancement and weights optimization. Input and output patterns are encoded as spike trains of precisely timed spikes, and the network learns to transform the input trains into target output trains. With these enhancements, the proposed methods outperformed other conventional neural network architectures.

In the paper entitled "*LGMS-FOA: an improved fruit fly optimization algorithm for solving optimization problems*," D. Shan et al. empirically study the performance of fruit

fly optimization algorithm (FOA). Six different nonlinear functions are selected as testing functions. The experimental results illustrate that FOA cannot solve complex optimization problems effectively. In order to enhance the performance of FOA, an improved FOA (named LGMS-FOA) is proposed. Simulation results and comparisons of LGMS-FOA with FOA and other metaheuristics show that LGMS-FOA can greatly enhance the searching efficiency and greatly improve the searching quality.

In another paper, Y. Wang et al. present “*Robust quadratic regression and its application to energy-growth consumption problem.*” The paper proposes a robust quadratic regression model to handle the statistics inaccuracy. Unlike the traditional robust statistic approaches that mainly focus on eliminating the effect of outliers, the proposed model employs the recently developed robust optimization methodology and tries to minimize the worst-case residual errors. First, they give a solvable equivalent semidefinite programming for the robust least square model with ball uncertainty set. Then the result is generalized to robust models under  $l_1$ - and  $l_\infty$ -norm criteria with general ellipsoid uncertainty sets. In addition, they establish a robust regression model for per capita GDP and energy consumption in the energy-growth problem under the conservation hypothesis. Finally, numerical experiments are carried out to verify the effectiveness of the proposed models and demonstrate the effect of the uncertainty perturbation on the robust models.

In another paper, the research of L. Wang entitled “*Extraction of belief knowledge from a relational database for quantitative Bayesian network inference,*” proposes the integration of a Bayesian network (BN) with the functional dependency (FD) discovery technique based on a three-phase learning framework. Association rule analysis is employed to discover FDs and expert knowledge encoded within a BN; that is, key relationships between attributes are emphasized. Moreover, the BN can be updated by using an expert-driven annotation process wherein redundant nodes and edges are removed. Experimental results show the effectiveness and efficiency of the proposed approach.

In the paper “*Identification of code-switched sentences and words using language modeling approaches,*” L.-C. Yu et al. propose a language modeling approach to the problem of code-switching language processing, dividing the problem into two subtasks: the detection of code-switched sentences and the identification of code-switched words in sentences. A code-switched sentence is detected on the basis of whether it contains words or phrases from another language. Once the code-switched sentences are identified, the positions of the code-switched words in the sentences are then identified. Experimental results show that the language modeling approach achieved an  $F$ -measure of 80.43% and an accuracy of 79.01% for detecting Mandarin-Taiwanese code-switched sentences. For the identification of code-switched words, the word-based and POS-based models achieved  $F$ -measures of 41.09% and 53.08%, respectively.

In the paper entitled “*Knowledge mining based on environmental simulation applied to wind farm power forecasting,*” D. Niu et al. propose a self-organizing map (SOM) combined with rough set theory clustering technique (RST) to extract

the relative knowledge and to choose the most similar history situation and efficient data for wind power forecasting with numerical weather prediction (NWP). Through integrating the SOM and RST methods to cluster the historical data into several classes, the approach could find the similar days and excavate the hidden rules. According to the data reprocessing, the selected samples will improve the forecast accuracy echo state network (ESN) trained by the class of the forecasting day that is adopted to forecast the wind power output accordingly. The developed methods are applied to a case of power forecasting in a wind farm located in northwest of China with wind power data from April 1, 2008, to May 6, 2009. In order to verify its effectiveness, the performance of the proposed method is compared with the traditional back-propagation neural network (BP). The results demonstrated that knowledge mining led to a promising improvement in the performance for wind farm power forecasting.

In the paper entitled “*Matching cost filtering for dense stereo correspondence,*” Y. Lin et al. propose a new cost-aggregation module to compute the matching responses for all the image pixels at a set of sampling points generated by a hierarchical clustering algorithm. The complexity of this implementation is linear both in the number of image pixels and in the number of clusters. Experimental results demonstrate that the proposed algorithm outperforms state-of-the-art local methods in terms of both accuracy and speed. Moreover, performance tests indicate that parameters such as the height of the hierarchical binary tree and the spatial and range standard deviations have a significant influence on time consumption and the accuracy of disparity maps.

In the paper entitled “*A multilayer hidden Markov models-based method for human-robot interaction,*” C. Tao and G. Liu propose a continuous gesture recognition approach based on multilayer hidden Markov models (MHMMs) to achieve human-robot interaction (HRI) by using gestures. The method consists of two parts. One part is gesture spotting and segment module, the other part is continuous gesture recognition module. Firstly, a Kinect sensor is used to capture 3D acceleration and 3D angular velocity data of hand gestures. Then, feed-forward neural networks (FNNS) and a threshold criterion are used for gesture spotting and segment, respectively. Afterwards, the segmented gesture signals are, respectively, preprocessed and vector symbolized by a sliding window and a K-means clustering method. Finally, symbolized data are sent into lower hidden Markov models (LHMMs) to identify individual gestures, and, then, a Bayesian filter with sequential constraints among gestures in upper hidden Markov models (UHMMs) is used to correct recognition errors created in LHMMs. Five predefined gestures are used to interact with a Kinect mobile robot in experiments. The experimental results show that the proposed method not only has good effectiveness and accuracy, but also has favorable real-time performance.

In the paper entitled “*Nonlinear predictive control of mass moment aerospace vehicles based on ant colony genetic algorithm optimization,*” X. Zhang et al. propose a novel kind of NPC parameters optimization strategy based on ant colony genetic algorithm (ACGA), aiming at the parameters of NPC that is generally used the trial-and-error method to optimize

and design. The method for setting NPC parameters with ACA in which the routes of ants are optimized by the genetic algorithm (GA) is derived. And then, a detailed realized process of this method is also presented. Furthermore, this optimization algorithm of the NPC parameters is applied to the flight control system of MMAV. The simulation results show that the system not only meets the demands of time-response specifications but also has excellent robustness.

In the paper entitled “*Modeling and analysis of the weld bead geometry in submerged arc welding by using adaptive neurofuzzy inference system*,” N. Akkas et al. aim at obtaining a relationship between the values defining bead geometry and the welding parameters and also to select optimum welding parameters. For this reason, an experimental study has been realized. The welding parameters such as the arc current, arc voltage, and welding speed that have the most effect on bead geometry are considered, and the other parameters are held as constant. Four, three, and five different values for the arc current, the arc voltage, and welding speed are used, respectively. Therefore, sixty samples made of St 52-3 material are prepared. The bead geometries of the samples are analyzed, and the thickness and penetration values of the weld bead are measured. Then, the relationship between the welding parameters is modeled by using artificial neural network (ANN) and neurofuzzy system approach. Each model is checked for its adequacy by using test data that are selected from experimental results. Then, the models developed are compared with regard to accuracy. In addition, the appropriate welding parameters values can be easily selected when the models improve.

In the paper entitled “*Genetic pattern search and its application to brain image classification*,” Y. Zhang et al. propose a novel global optimization method based on the combination of genetic algorithm (GA) and generalized pattern search (PS) algorithm, to find global minimal points more effectively and rapidly. The idea lies in the facts that GA tends to be quite good at finding generally good global solutions but quite inefficient in finding the last few mutations for the absolute optimum and that PS is quite efficient in finding absolute optimum in a limited region. The novel algorithm, named as genetic pattern search (GPS), employs the GA as the search method at every step of PS. Experiments on five different classical benchmark functions (consisting of Hump, Powell, Rosenbrock, Schaffer, and Woods) demonstrate that the proposed GPS is superior to improved GA and improved PS with respect to success rate. They apply the GPS to the classification of normal and abnormal structural brain MRI images. The results indicate that GPS exceeds BP, MBP, IGA, and IPS in terms of classification accuracy. This suggests that GPS is an effective and viable global optimization method and can be applied to brain MRI classification.

In the paper entitled “*Solving a novel inventory location model with stochastic constraints and (R, s, S) inventory control policy*,” G. Cabrera et al. solve a novel inventory-location model with a stochastic capacity constraint based on a periodic inventory control (ILM-PR) policy. The ILM-PR policy implies several changes with regard to other previous models proposed in the literature, which consider continuous review as their inventory policy. One of these changes is

the inclusion of the undershoot concept, which has not been considered in previous ILM models in the literature. Based on the model, they design a distribution network for a two-level supply chain, addressing both warehouse location and customer assignment decisions, whilst taking into consideration several aspects of inventory planning, in particular, evaluating the impact of the inventory control review period on the network configuration and system costs. Because the model is a very hard-to solve combinatorial nonlinear optimization problem, they implemented two heuristics to solve it, namely, Tabu Search and Particle Swarm Optimization. These approaches are tested over small instances in which they are able to find the optimal solution in just a few seconds. Because the model is a new one, a set of medium-size instances is provided that can be useful as a benchmark in future research. The heuristics show a good convergence rate when applied to those instances. The results confirm that decision making over the inventory control policy has effects on the distribution network design.

In another paper, S. Dinç et al. present “*Hyperspectral image classification using kernel Fukunaga-Koontz transform*.” The paper presents a novel approach for the hyperspectral imagery (HSI) classification problem, using Kernel Fukunaga-Koontz transform (K-FKT). The Kernel based Fukunaga-Koontz transform offers higher performance for classification problems due to its ability to solve nonlinear data distributions. K-FKT is realized in two stages: training and testing. In the training stage, unlike classical FKT, samples are relocated to the higher dimensional kernel space to obtain a transformation from nonlinear distributed data to linear form. This provides a more efficient solution to hyperspectral data classification. The second stage, testing, is accomplished by employing the Fukunaga-Koontz transformation operator to find out the classes of the real world hyperspectral images. In experiment section, the improved performance of HSI classification technique, K-FKT, has been tested comparing other methods such as the classical FKT and three types of support vector machines (SVMs).

In another paper, the research of A. El Mobacher et al. entitled “*Entropy-based and weighted selective SIFT clustering as an energy aware framework for supervised visual recognition of man-made structures*,” presents uSee, a supervised learning framework which exploits the symmetrical and repetitive structural patterns in buildings to identify subsets of relevant clusters formed by these keypoints. Once a smart phone captures an image, uSee preprocesses it using variations in gradient angle- and entropy-based measures before extracting the building signature and comparing its representative SIFT keypoints against a repository of building images. Experimental results on 2 different databases confirm the effectiveness of uSee in delivering, at a greatly reduced computational cost, the high matching scores for building recognition that local descriptors can achieve. With only 14.3% of image SIFT keypoints, uSee exceeded prior literature results by achieving an accuracy of 99.1% on the Zurich Building Database with no manual rotation, thus saving significantly on the computational requirements of the task at hand.

In the paper “*Application of adaptive extended Kalman smoothing on INS/WSN integration system for mobile robot indoors*,” X. Chen et al. propose the inertial navigation systems (INS)/wireless sensor network (WSN) integration system of mobile robot for navigation information indoors accurately and continuously. The Kalman filter (KF) is widely used for real-time applications with the aim of gaining optimal data fusion. In order to improve the accuracy of the navigation information, they propose an adaptive extended Kalman smoothing (AEKS) which utilizes inertial measuring units (IMUs) and ultrasonic positioning system. The adaptive extended Kalman filter (AEKF) is used to improve the accuracy of forward Kalman filtering (FKF) and backward Kalman filtering (BKF), and then the AEKS and the average filter are used between two output timings for the online smoothing. Several real indoor tests are done to assess the performance of the proposed method. The results show that the proposed method can reduce the error compared with the INS-only, least squares (LS) solution, and AEKF.

In the paper entitled “*The study of reinforcement learning for traffic self-adaptive control under multiagent Markov game environment*,” L.-H. Xu et al. cast traffic self-adaptive control as a multiagent Markov game problem. The design employs traffic signal control agent (TSCA) for each signalized intersection that coordinates with neighboring TSCAs. A mathematical model for TSCAs’ interaction is built based on nonzero-sum Markov game that has been applied to let TSCAs learn how to cooperate. A multiagent Markov game reinforcement learning approach is constructed on the basis of single-agent Q-learning. Their method lets each TSCA learn to update its Q-values under the joint actions and imperfect information. The convergence of the proposed algorithm is analyzed theoretically. The simulation results show that the proposed method is convergent and effective in realistic traffic self-adaptive control setting.

In the paper entitled “*Research on ISFLA-based optimal control strategy for the coordinated charging of EV battery swap station*,” X. Huang et al. maintain that electric vehicles (EVs) could be a good measure against energy shortages and environmental pollution as an important component of the smart grid. A main way of energy supply to EVs is to swap battery from the swap station. Based on the characteristics of EV battery swap station, the coordinated charging optimal control strategy is investigated to smooth the load fluctuation. Shuffled frog leaping algorithm (SFLA) is an optimization method inspired by the memetic evolution of a group of frogs when seeking food. An improved shuffled frog leaping algorithm (ISFLA) with the reflecting method to deal with the boundary constraint is proposed to obtain the solution of the optimal control strategy for coordinated charging. Based on the daily load of a certain area, the numerical simulations including the comparison of PSO and ISFLA are carried out and the results show that the presented ISFLA can effectively lower the peak-valley difference and smooth the load profile with the faster convergence rate and higher convergence precision.

In the paper entitled “*Composite broadcasting and ranging via a satellite dual-frequency MPPSK system*,” Y. Yao et al. propose the design of dual frequency M-ray position phase shift

keying (MPPSK) system that is suitable for performing both data transmission and range measurement. The approach is based on MPPSK modulation waveforms utilized in digital video broadcasting. In particular, requirements that allow for employing such signals for range measurements with high accuracy and high range are investigated. In addition, the relationship between the frequency difference of dual frequency MPPSK system and range accuracy is discussed. Moreover, the selection of MPPSK modulation parameter for data rate and ranging is considered. In addition to theoretical considerations, the paper presents system simulations and measurement results of new systems, demonstrating the high spectral utilization of integrated broadcasting and ranging applications.

In the paper entitled “*A genetic-algorithms-based approach for programming linear and quadratic optimization problems with uncertainty*,” W. Jin et al. propose a genetic-algorithms-based approach as an all-purpose problem-solving method for operation programming problems under uncertainty. The proposed method is applied for management of a municipal solid waste treatment system. Compared to the traditional interactive binary analysis, this approach has fewer limitations and is able to reduce the complexity in solving the inexact linear programming problems and inexact quadratic programming problems. The implementation of this approach is performed using the genetic algorithm solver of MATLAB (trademark of MathWorks). The paper explains the genetic-algorithms-based method and presents details on the computation procedures for each type of inexact operation programming problems. A comparison of the results generated by the proposed method based on genetic algorithms with those produced by the traditional interactive binary analysis method is also presented.

In the paper entitled “*Solving the balanced academic curriculum problem using the ACO metaheuristic*,” J.-M. Rubio et al. consider that the balanced academic curriculum problem consists in the assignation of courses to academic periods satisfying all the load limits and prerequisite constraints. They present the design of a solution to the balanced academic curriculum problem based on the ACO metaheuristic, in particular via the best-worst ant system. They provide an experimental evaluation that illustrates the effectiveness of the proposed approach on a set of classic benchmarks as well as on real instances.

In the paper entitled “*Hybrid functional-neural approach for surface reconstruction*,” A. Iglesias and A. Gálvez introduce a new hybrid functional-neural approach for surface reconstruction. The approach is based on the combination of two powerful artificial intelligence paradigms: on one hand, they apply the popular Kohonen neural network to address the data parameterization problem. On the other hand, they introduce a new functional network, called NURBS functional network, whose topology is aimed at reproducing faithfully the functional structure of the NURBS surfaces. These neural and functional networks are applied in an iterative fashion for further surface refinement. The hybridization of these two networks provides a powerful computational approach to obtain a NURBS fitting surface

to a set of irregularly sampled noisy data points within a prescribed error threshold. The method has been applied to two illustrative examples. The experimental results confirm the good performance of their approach.

In the paper entitled “*An algorithm for mining of association rules for the information communication network alarms based on swarm intelligence*,” Y. Wang et al. analyze the data characteristics and association logic of the network alarms. Besides, the alarm data are preprocessed and the main standardization information fields are screened. The APPSO algorithm is proposed on the basis of combining the evaluation method for support and confidence coefficient in the Apriori (AP) algorithm as well as the particle swarm optimization (PSO) algorithm. By establishing a sparse linked list, the algorithm is able to calculate the particle support thus further improving the performance of the APPSO algorithm. Based on the test for the network alarm data, it is discovered that rational setting of the particle swarm scale and number of iterations of the APPSO algorithm can be used to mine the vast majority and even all of the association rules and the mining efficiency are significantly improved, compared with Apriori algorithm.

In another paper, S.-F. Jiang et al. present “*Structural reliability assessment by integrating sensitivity analysis and support vector machine*.” This paper proposes a structural reliability assessment method by the use of sensitivity analysis (SA) and support vector machine (SVM). The sensitivity analysis is firstly applied to assess the effect of random variables on the values of performance function, while the small-influence variables are rejected as input vectors of SVM. Then, the trained SVM is used to classify the input vectors, which are produced by sampling the residual variables based on their distributions. Finally, the reliability assessment is implemented with the aid of reliability theory. A 10-bar planar truss is used to validate the feasibility and efficiency of the proposed method, and a performance comparison is made with other existing methods. The results show that the proposed method can largely save the runtime with less reduction of the accuracy; furthermore, the accuracy using the proposed method is the highest among the methods employed.

In a paper, the research of C. Liu et al. entitled “*Incremental tensor principal component analysis for handwritten digit recognition*” proposes incremental tensor principal component analysis (ITPCA) based on updated-SVD technique algorithm. This paper proves the relationship between PCA, 2DPCA, MPCA, and the graph embedding framework theoretically and derives the incremental learning procedure to add single sample and multiple samples in detail. The experiments on handwritten digit recognition have demonstrated that ITPCA has achieved better recognition performance than that of vector-based principal component analysis (PCA), incremental principal component analysis (IPCA), and multilinear principal component analysis (MPCA) algorithms. At the same time, ITPCA also has lower time and space complexity.

In the paper “*Optimum performance-based seismic design using a hybrid optimization algorithm*,” S. Talatahari et al. present a hybrid optimization method to optimum seismic

design of steel frames considering four performance levels. These performance levels are considered to determine the optimum design of structures to reduce the structural cost. A pushover analysis of steel building frameworks subject to equivalent-static earthquake loading is utilized. The algorithm is based on the concepts of the charged system search in which each agent is affected by local and global best positions stored in the charged memory considering the governing laws of electrical physics. Comparison of the results of the hybrid algorithm with those of other metaheuristic algorithms shows the efficiency of the hybrid algorithm.

In the paper entitled “*Research on cooperative combat for integrated reconnaissance-attack-BDA of group LAVs*,” L. Bing et al. analyze a group system by the theory of Itô stochastic differential. The uniqueness and continuity of the solution of the system are discussed. Afterwards the model of the system based on the state transition is established with the finite state machine automatically. At last, a search algorithm is proposed for obtaining good feasible solutions for problems. Moreover, simulation results show that model and method are effective for dealing with cooperative combat of group LAVs (loitering air vehicles).

In the paper entitled “*Wolf pack algorithm for unconstrained global optimization*,” H.-S. Wu and F.-M. Zhang abstract three intelligent behaviors, scouting, calling, and besieging, and two intelligent rules, winner-take-all generation rule of lead wolf and stronger-survive renewing rule of wolf pack. Then they propose a new heuristic swarm intelligent method, named wolf pack algorithm (WPA). Experiments are conducted on a suit of benchmark functions with different characteristics, unimodal/multimodal, separable/nonseparable, and the impact of several distance measurements and parameters on WPA are discussed. What is more, the compared simulation experiments with other five typical intelligent algorithms, genetic algorithm, particle swarm optimization algorithm, artificial fish swarm algorithm, artificial bee colony algorithm, and firefly algorithm, show that WPA has better convergence and robustness, especially for high-dimensional functions.

In the paper entitled “*Towards a unified sentiment lexicon based on graphics processing units*,” L. I. Barbosa-Santillán et al. present an approach to create what they have called a unified sentiment lexicon (USL). This approach aims at aligning, unifying, and expanding the set of sentiment lexicons that are available on the web in order to increase their robustness of coverage. One problem related to the task of the automatic unification of different scores of sentiment lexicons is that there are multiple lexical entries for which the classification of positive, negative, or neutral ( $P$ ,  $N$ ,  $Z$ ) depends on the unit of measurement used in the annotation methodology of the source sentiment lexicon. Their USL approach computes the unified strength of polarity of each lexical entry based on the Pearson correlation coefficient that measures how correlated lexical entries are with a value between 1 and  $-1$ , where 1 indicates that the lexical entries are perfectly correlated, 0 indicates no correlation, and  $-1$  means they are perfectly inversely correlated and so is the unified metrics procedure for CPU and GPU, respectively. Another problem is the high processing time required for computing

all the lexical entries in the unification task. Thus, the USL approach computes a subset of lexical entries in each of the 1344 GPU cores and uses parallel processing in order to unify 155802 lexical entries. The results of the analysis conducted using the USL approach show that the USL has 95.430 lexical entries, out of which there are 35.201 considered positive, 22.029 negative, and 38.200 neutral. Finally, the runtime is 10 minutes for 95.430 lexical entries; this allows a reduction of the time computing for the unified metrics by 3 times.

In the paper entitled “*The application of FastICA combined with related function in blind signal separation*,” D. Li et al. propose a new method which is the combination of related function relevance to estimated signal and negative entropy in fast independent component analysis (FastICA) as objective function, and the iterative formula is derived without any assumptions, then the independent components are found by maximizing the objective function. The improved algorithm shorthand for R-FastICA is applied to extract random mixed signals and ventricular late potential (VLP) signal from normal ECG signal; simultaneously the performance of R-FastICA algorithm is compared with traditional FastICA through simulation. Experimental results show that R-FastICA algorithm outperforms traditional FastICA with higher similarity coefficient and separation precision.

## Acknowledgments

We would like to express our gratitude to all of the authors for their contributions and the reviewers for their effort providing constructive comments and feedback. We hope this special issue offers a comprehensive and timely view of the area of emerging trends in artificial intelligence and its applications and that it will offer stimulation for further research.

*Yudong Zhang  
Saeed Balochian  
Praveen Agarwal  
Vishal Bhatnagar  
Orwa Jaber Housheya*

## Research Article

# The Application of FastICA Combined with Related Function in Blind Signal Separation

**Dengao Li, Junmin Zhao, Hongyan Liu, and Defeng Hao**

*College of Information Engineering, Taiyuan University of Technology, Taiyuan 030024, China*

Correspondence should be addressed to Dengao Li; [lidengaoty@126.com](mailto:lidengaoty@126.com)

Received 20 July 2013; Accepted 7 March 2014; Published 3 April 2014

Academic Editor: Vishal Bhatnagar

Copyright © 2014 Dengao Li et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Blind source separation (BSS) has applications in the fields of data compression, feature recognition, speech, audio, and biosignal processing. Identification of ECG signal is one of the challenges in the biosignal processing. Proposed in this paper is a new method, which is the combination of related function relevance to estimated signal and negative entropy in fast independent component analysis (FastICA) as objective function, and the iterative formula is derived without any assumptions; then the independent components are found by maximizing the objective function. The improved algorithm shorthand for R-FastICA is applied to extract random mixed signals and ventricular late potential (VLP) signal from normal ECG signal; simultaneously the performance of R-FastICA algorithm is compared with traditional FastICA through simulation. Experimental results show that R-FastICA algorithm outperforms traditional FastICA with higher similarity coefficient and separation precision.

## 1. Introduction

Blind source separation (BSS) [1] has been applied successfully to extract mixed signals in different fields of data compression [2], feature recognition [3], speech, audio, and biosignal processing [4, 5] as a statistical signal method. Literature [2] showed that the compression ratio was higher through the ICA method than principal component analysis (PCA). The accuracy ratio of feature recognition was 92.1% based on the complex valued Independent Component Analysis in literature [3]. Literature [4, 5] showed that FastICA was an efficient method in the extraction of speech, audio, and biosignal. Neither the source signals nor the structure of mixed matrix is known [6].

The detection and analysis of VLP generally appearing in the end of QRS wave and extending to ST segment with a series of high frequency and low-rising weak irregular electrical signal are a kind of effective means to predict unexplained asphyxia, sudden cardiac deaths, and so forth [7]. At present, the analysis methods of VLP commonly have had a time domain method, frequency domain method, spectrum scale measurement analysis method, and so forth [8]. Time domain method is not easy to improve detection rate of VLP;

frequency domain analysis is limited by frequency resolution; spectrum scale measurement analysis can overcome some limitations in time domain and frequency domain analysis, but it is easily influenced by the selection of analysis time and positioning of QRS wave terminal in extracting judgmental standard parameters of VLP [9].

To overcome the abovementioned limitation and improve the detection accuracy, it is necessary to put forward a new detection technology. Independent component analysis (ICA) as a branch of BSS is widely applied to this problem in recent years [10–12]. Traditional FastICA algorithm has obtained several effects in extracting electromyographic and atrial fibrillation signal. VLP compared with normal ECG signal waveform has a relative independence, and ICA algorithm can accurately distinguish relatively independent component from the ECG; it can be used to identify VLP. In addition, many authors have conducted specific research to ICA algorithm. Literature [13] proposed an adaptive ICA algorithm based on artificial neural network, which reduced the complexity of obtaining the learning matrix and independent component; literature [14] designed a fast search algorithm directly based on kurtosis as a measure of non-Gauss; literature [15] analyzed the principle and method of

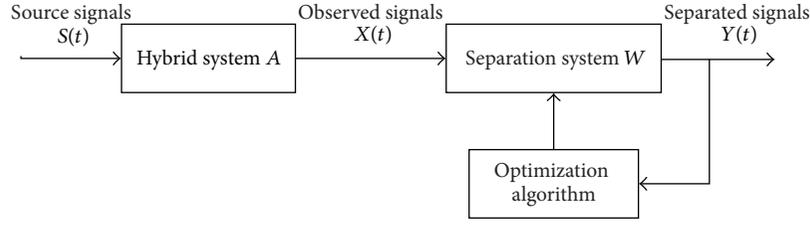


FIGURE 1: FastICA principle diagram.

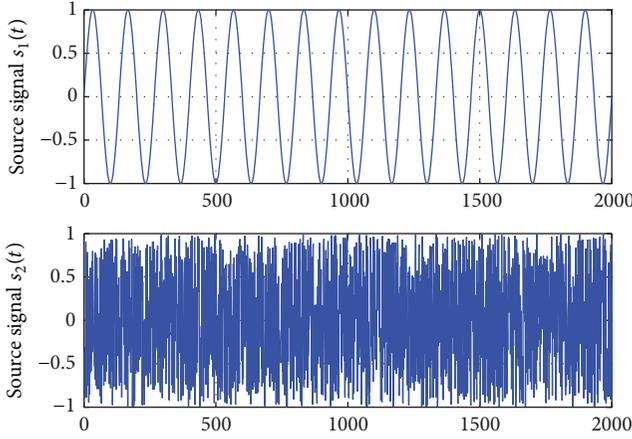


FIGURE 2: The source signals.

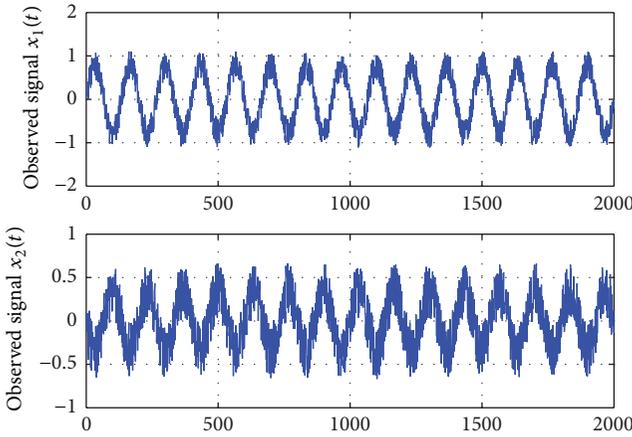


FIGURE 3: The mixed signals.

independent component in serial estimation ICA model; the concept of kernel ICA was proposed in literature [16], where data was analyzed through related analysis combined with mutual information theory; it had obtained better separation effect.

R-FastICA is proposed in this paper, which is the combination of related function and negative entropy as objective function, and the iterative formula is derived; then the independent components are found by maximizing the objective function. The extracted performance of R-FastICA algorithm is compared with traditional FastICA through simulation of

random mixed signals and ECG signal with VLP. Experimental results show that R-FastICA algorithm outperforms traditional FastICA with higher similarity coefficient and separation precision.

## 2. Theory

ICA firstly proposed by Pierre Comon in 1994 is a method for finding the statistical independent components from multidimensional statistical data [17]. The mathematical model without noises can be expressed as follows:

$$X(t) = AS(t), \quad (1)$$

where  $S(t) = [s_1(t), s_2(t), \dots, s_N(t)]^T$  is a  $N$  column vector of source signals and  $X(t) = [x_1(t), x_2(t), \dots, x_M(t)]^T$  is  $M$  column vector of observed signals.  $A$  is a  $M \times N$  mixed matrix required that  $M \geq N$ .

The goal is to extract independent source signals from mixed signals by finding separation matrix  $W$  through some assumptions and constraints under the premise of unknown source signal  $S(t)$  and mixed matrix  $A$ , which makes output  $Y(t)$  an estimation to source signal  $S(t)$ . That is to say,

$$Y(t) = WX(t) = WAS(t) \approx S(t), \quad (2)$$

where  $Y(t) = [y_1(t), y_2(t), \dots, y_N(t)]^T$  is the estimation of source signals.

Generally, the normal ECG and VLP signal can be thought of as statistical independence with each other; thus VLP signal will be extracted through FastICA algorithm [11].

The basic model is shown in Figure 1.

## 3. Methodology

**3.1. R-FastICA Algorithm.** Traditional FastICA method is to estimate source signals based on negative entropy.

In order to improve separation precisely, negative entropy combined with related function as objective function is proposed in this paper. The updating formula of R-FastICA algorithm is vector gradient derived by the negative entropy combined with related function. The basic idea of R-FastICA algorithm requires that extracted signals are not only independent but also have high precision. Related function  $\delta$  is introduced and defined by following formula:

$$\delta = \lg \frac{\sum_{i=1}^N y_i \cdot y_i^T}{\text{MSE}}, \quad (3)$$

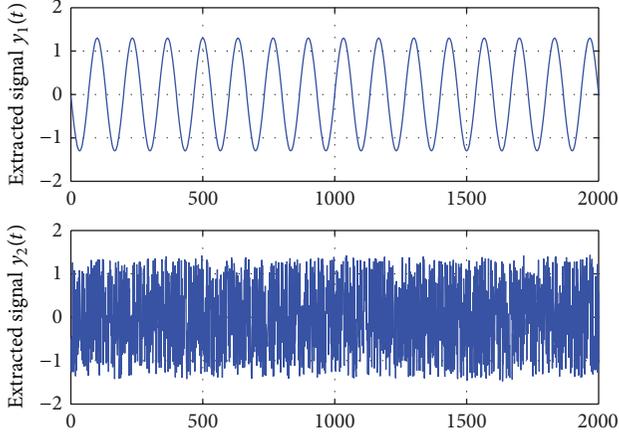


FIGURE 4: Extracted signals with R-FastICA.

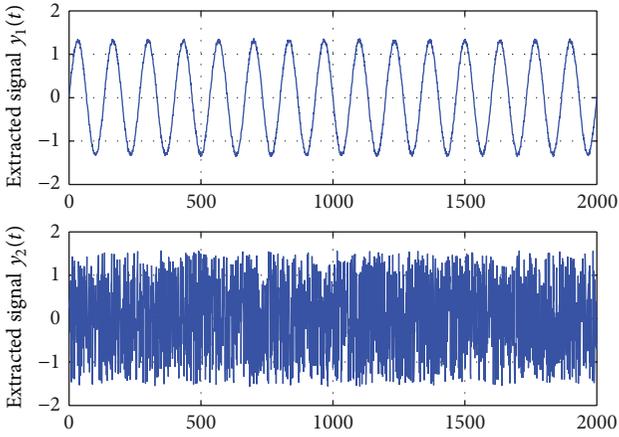


FIGURE 5: Extracted signals with traditional FastICA.

where MSE represents the mean square error between source signal and estimated signal, defined as [18]

$$\text{MSE} = \sum_{i=1}^N \frac{(s_i - y_i)(s_i - y_i)^T}{N}. \quad (4)$$

The MSE in type (3) is substituted by type (4), simplified as

$$\delta = N \lg \frac{y \cdot y^T}{(s - y)(s - y)^T}. \quad (5)$$

Source signal is replaced by the average of estimated signal. It is defined by  $s = y^* = (1/p) \sum_{j=1}^p y(t - j)$ , where  $p$  is an arbitrary integer less than 100 [19]. The type (5) can be simplified as

$$\delta = N \lg \frac{y \cdot y^T}{(y^* - y)(y^* - y)^T}. \quad (6)$$

BSS model is  $Y(t) = WX(t)$  and  $X^*$  is the average of  $X$ . It can be known that

$$y^* = w^T X^*, \quad (7)$$

where  $w$  is a column vector and  $w^T$  is a row vector.

TABLE 1: The comparison of similarity coefficient matrix.

	R-FastICA		FastICA	
Similarity	-1.0000	0.0001	0.9996	0.0003
coefficient matrix	0.0003	-1.0000	0.0005	0.9997

TABLE 2: The comparison of similarity coefficient matrix.

	R-FastICA		FastICA	
Similarity	1.0000	0.0002	0.9995	-0.0006
coefficient matrix	0.0003	0.9999	0.0004	-0.9994

Combining type (6) and type (7), it can be got that

$$\begin{aligned} \delta &= \lg \frac{w^T X \cdot (w^T X)^T}{(w^T X^* - w^T X)(w^T X^* - w^T X)^T} \\ &= \lg \frac{w^T X(X)^T w}{w^T (X^* - X)(X^* - X)^T w} \\ &= \lg \frac{U}{V} = \lg U - \lg V, \end{aligned} \quad (8)$$

where

$$U = w^T X(X)^T w$$

$$= [w_1 w_2 \cdots w_M] \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_M \end{bmatrix} [x_1 \ x_2 \ \cdots \ x_M] \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_M \end{bmatrix} \quad (9)$$

$$\begin{aligned} &= w_1^2 x_1^T x_1 + w_2^2 x_2^T x_2 + \cdots + w_M^2 x_M^T x_M \\ &= \sum_{i=1}^M \sum_{j=1}^M w_i w_j x_i^T x_j. \end{aligned}$$

Similarly,

$$\begin{aligned} V &= w^T (X^* - X)(X^* - X)^T w \\ &= \sum_{i=1}^M \sum_{j=1}^M w_i w_j (x_i^* - x_i)^T (x_j^* - x_j). \end{aligned} \quad (10)$$

From the above,  $\delta$  is a function based on  $w$  and  $X$ ; the vector gradient can be obtained by the bottom of related function:

$$\frac{\partial F(w, X)}{\partial w} = \frac{\partial(\delta)}{\partial w} = \frac{1}{U} \frac{\partial U}{\partial w} - \frac{1}{V} \frac{\partial V}{\partial w}. \quad (11)$$

Vector gradient is defined as

$$\frac{\partial g}{\partial w} = \left[ \frac{\partial g}{\partial w_1}, \frac{\partial g}{\partial w_2}, \cdots, \frac{\partial g}{\partial w_M} \right]^T. \quad (12)$$

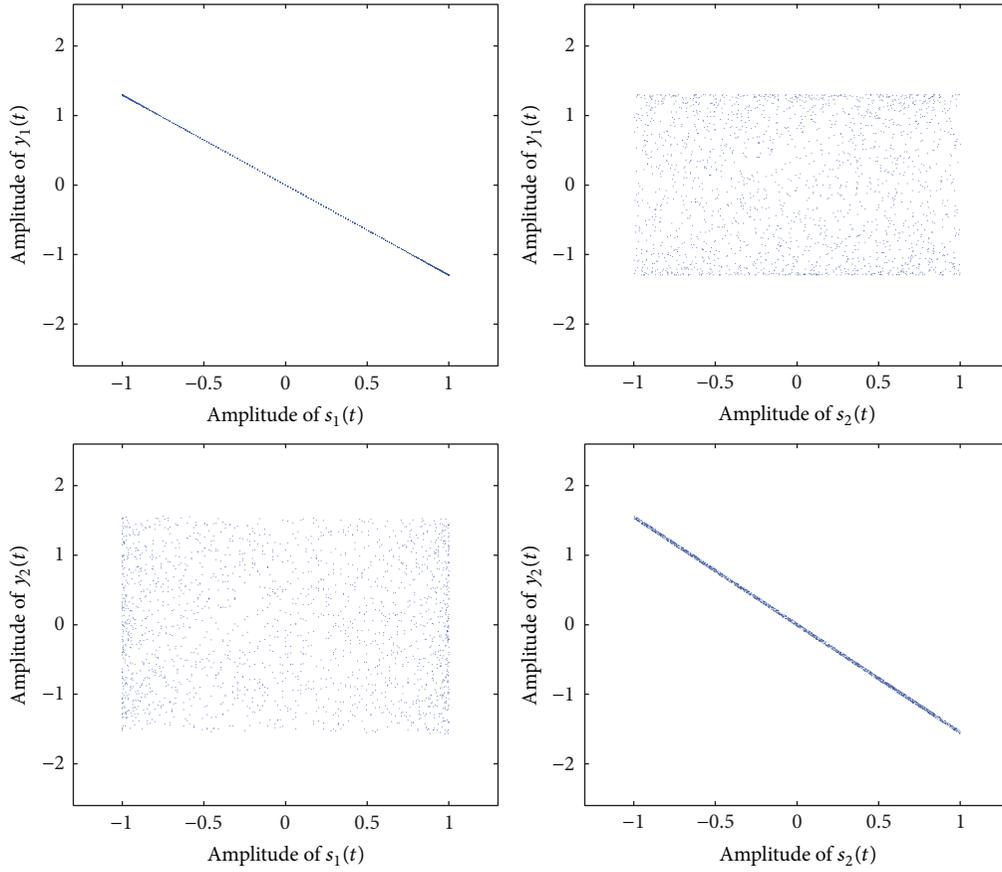


FIGURE 6: The composite scattering plot with R-FastICA algorithm.

The gradient of  $U$  can be calculated according to type (12):

$$\begin{aligned}
 \frac{\partial U}{\partial w} &= \frac{\partial (\sum_{i=1}^M \sum_{j=1}^M w_i w_j x_i^T x_j)}{\partial w} \\
 &= 2 \left[ \sum_{j=1}^M w_j x_1^T x_j, \sum_{j=1}^M w_j x_2^T x_j, \dots, \sum_{j=1}^M w_j x_M^T x_j \right] \\
 &= \begin{bmatrix} x_1^T & x_1 & x_1^T & x_2 & \cdots & x_1^T & x_M \\ x_2^T & x_1 & x_2^T & x_2 & \cdots & x_1^T & x_M \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ x_M^T & x_1 & x_M^T & x_2 & \cdots & x_M^T & x_M \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_M \end{bmatrix} \quad (13) \\
 &= 2 \begin{bmatrix} x_1^T \\ x_2^T \\ \vdots \\ x_M^T \end{bmatrix} [x_1 \ x_2 \ \cdots \ x_M] \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_M \end{bmatrix} \\
 &= 2XX^T w.
 \end{aligned}$$

Similarly,

$$\begin{aligned}
 \frac{\partial V}{\partial w} &= \frac{\partial w^T (X^* - X)(X^* - X)^T w}{\partial w} \\
 &= 2(X^* - X)(X^* - X)^T w.
 \end{aligned} \quad (14)$$

Type (15) is a gradient of related function to  $w$ , which can be calculated combining type (13) with type (14):

$$\frac{\partial F(w, X)}{\partial w} = 2 \left\{ \frac{XX^T w}{w^T XX^T} - \frac{(X^* - X)(X^* - X)^T w}{w^T (X^* - X)(X^* - X)^T w} \right\}. \quad (15)$$

In literature [10], the approximate calculation formula of negative entropy is

$$J(y) \propto \{E[G(y)] - E[G(v)]\}^2. \quad (16)$$

The objective function is composed by negative entropy and related function including the information between source signal and estimated signal in R-FastICA:

$$\varphi(y) = \{E[G(y)] - [G(v)]\}^2 \cdot \delta, \quad (17)$$

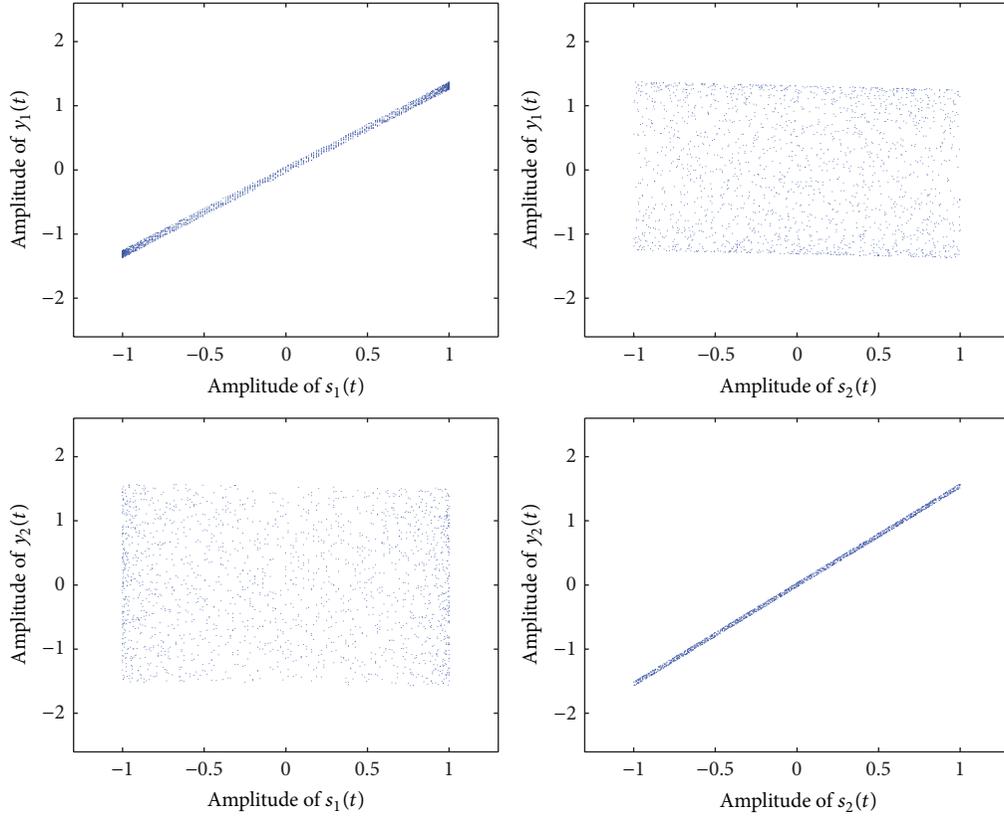


FIGURE 7: The composite scattering plot with FastICA algorithm.

where  $v$  and  $y$  are Gauss random variables with the same covariance (zero mean and unit variance) and  $G$  is a nonlinear function selected by distribution form of source signals.

The vector gradient of objective function  $\varphi(y)$  to  $w$  is as follows:

$$\begin{aligned}
 \frac{\partial \varphi(y)}{\partial w} &= \frac{\partial \{E[G(y)] - E[G(v)]\}^2 F(w, X)}{\partial w} \\
 &= 2 \frac{\partial \{E[G(y)] - E[G(v)]\}}{\partial w} \\
 &\quad \cdot F(w, X) + \{E[G(y)] - E[G(v)]\}^2 \cdot \frac{\partial F(w, X)}{\partial w} \\
 &= 2 \{E[Xg(w^T X)] - E[G(v)]\} \\
 &\quad \cdot \lg \frac{w^T X X^T w}{w^T (X^* - X)(X^* - X)^T w} \\
 &\quad + 2\{E[G(y)] - E[G(v)]\}^2 \\
 &\quad \cdot E \left[ \frac{X X^T w}{w^T X X^T w} - \frac{(X^* - X)(X^* - X)^T w}{w^T (X^* - X)(X^* - X)^T w} \right]. \tag{18}
 \end{aligned}$$

Type (18) is a new updating formula. The improved algorithm can ensure that the estimated signals are independent and the precision is higher due to the fact that related function  $\delta$  is relevant to estimated signal.

3.2. *Assessment Method.* A familiar measure of separation performance is the similarity coefficient defined as [20]

$$\varepsilon_{ij} = \varepsilon(y_i, s_j) = \frac{|\sum_{i,j=1}^N y_i(t) s_j(t)|}{\sqrt{\sum_{i=1}^N y_i^2(t) \sum_{j=1}^N s_j^2(t)}}. \tag{19}$$

When  $y_i = c s_j$ , separation effect is ideal; when  $y_i$  and  $s_j$  are mutual independent,  $\varepsilon_{ij} = 0$ ; generally, similarity coefficient matrix is used to measure extracted performance.

The composite scattering plot [21] is a measure to describe corresponding relationship between the source signal and the extracted signal, where we can see that not only an extracted signal is the recovery of a source signal, but also the phase of source signal and extracted signal is the same or opposite.

## 4. Simulation

In order to indicate the performance of R-FastICA compared with traditional FastICA, the following simulations were conducted.

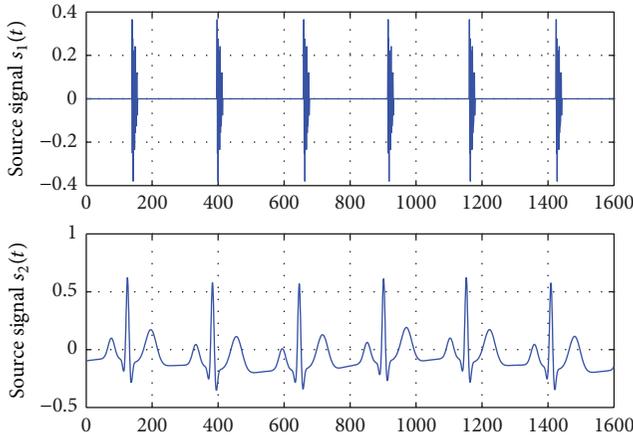


FIGURE 8: The source signals.

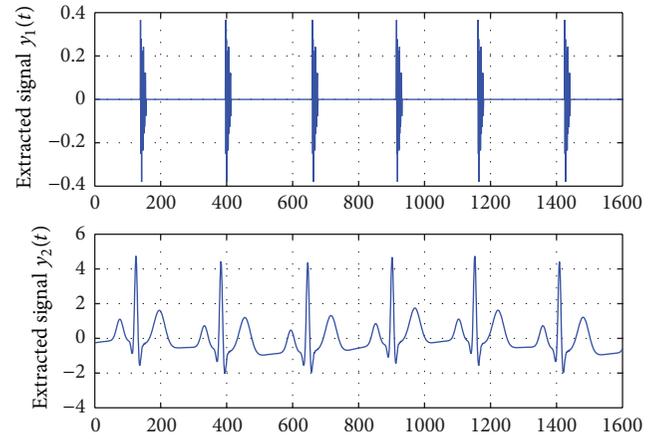


FIGURE 10: Extracted signals with R-FastICA.

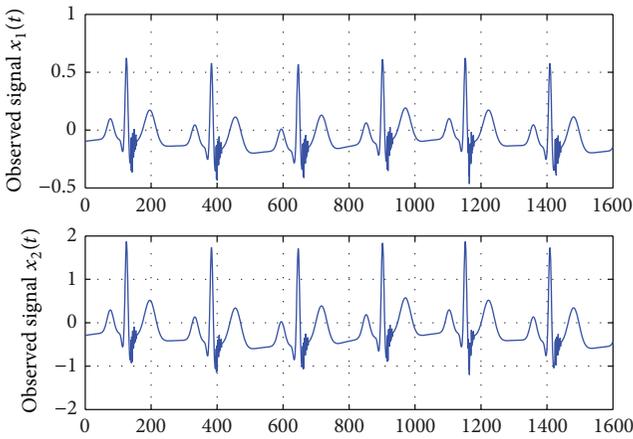


FIGURE 9: The observed signals.

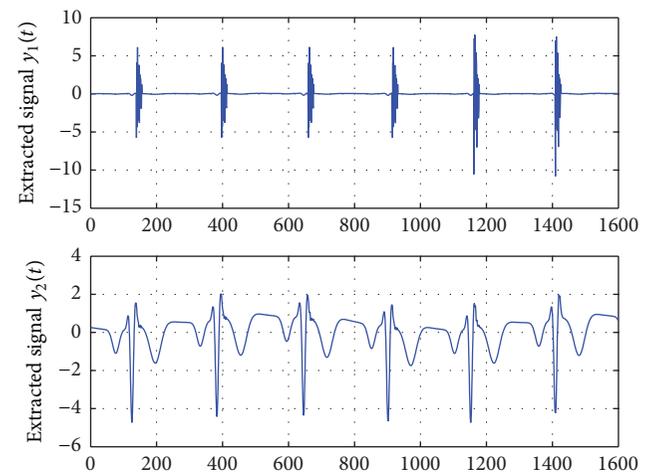


FIGURE 11: Extracted signals with FastICA.

Taking random signals as an example in the first simulation, R-FastICA method was proved to be effective. In the second simulation, taking ECG signal with VLP as an example, the original ECG signal without noises was from MIT/BIH database and VLP signal was generated through stacking sine waves with different frequency and amplitude [22].

**4.1. Source Separation of the Random Signals.** In this simulation, the source signal  $s_1(t)$  was sinusoidal signal and  $s_2(t)$  was random noises, whose sampling number was 2000. Signals  $s_1(t)$  and  $s_2(t)$  were shown in Figure 2, as well as mixed signals  $x_1(t)$  and  $x_2(t)$  that were shown in Figure 3.

The mixed signals were extracted through R-FastICA and traditional FastICA in Figures 4 and 5, respectively.

In the experiment of extracting random signals, the comparison of similarity coefficient matrix between RFastICA and FastICA algorithm was shown in Table 1.

Extracted signal  $y_1(t)$  was the estimation of source signal  $s_1(t)$  and the phase was opposite in Figure 6 and the same in Figure 7. Extracted signal  $y_2(t)$  was the estimation of source signal  $s_2(t)$  and the phase was opposite in Figure 6 and the same in Figure 7.

From the above experiments, we could see that R-FastICA algorithm outperforms traditional FastICA with higher similarity coefficient and separation precision.

**4.2. Source Separation of the VLP Signal.** In this simulation, the source signal  $s_1(t)$  was VLP signal generated by program and  $s_2(t)$  was ECG signal from MIT/BIH database, whose sampling number was 1600. The source signals  $s_1(t)$  and  $s_2(t)$ , as well as mixed signals  $x_1(t)$  and  $x_2(t)$ , were shown in Figures 8 and 9, respectively.

The mixed signals were extracted through R-FastICA and traditional FastICA in Figures 10 and 11, respectively.

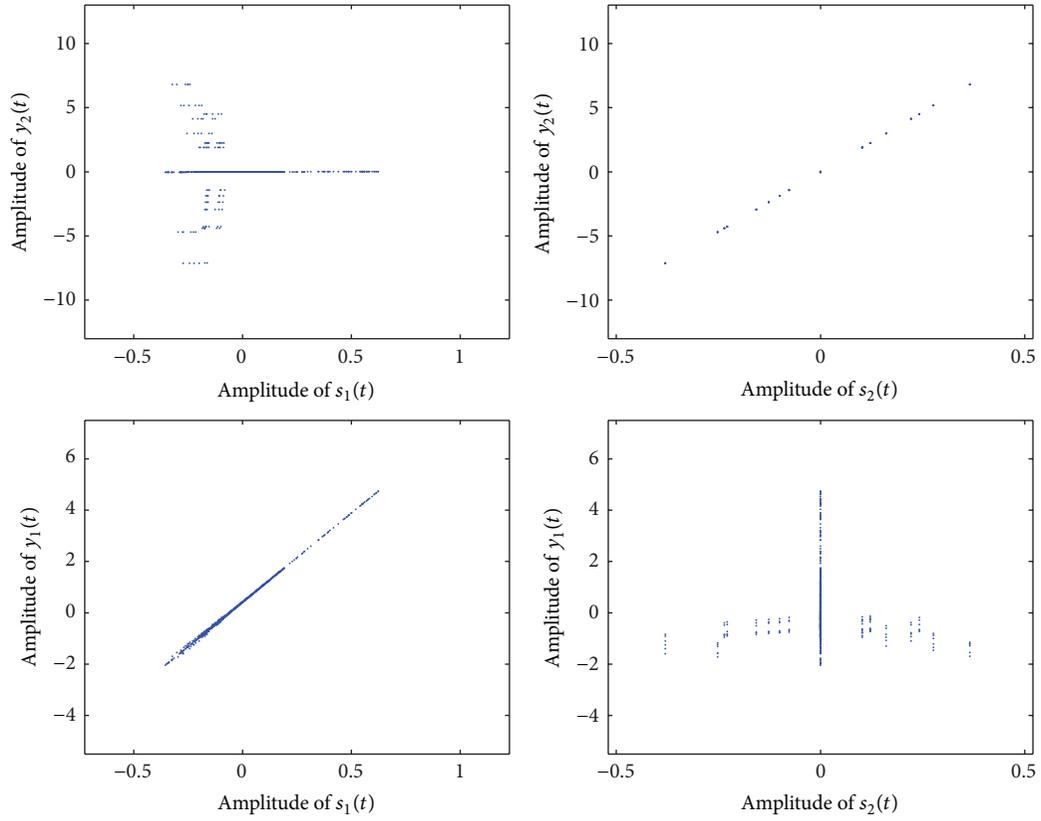


FIGURE 12: The composite scattering plot with R-FastICA.

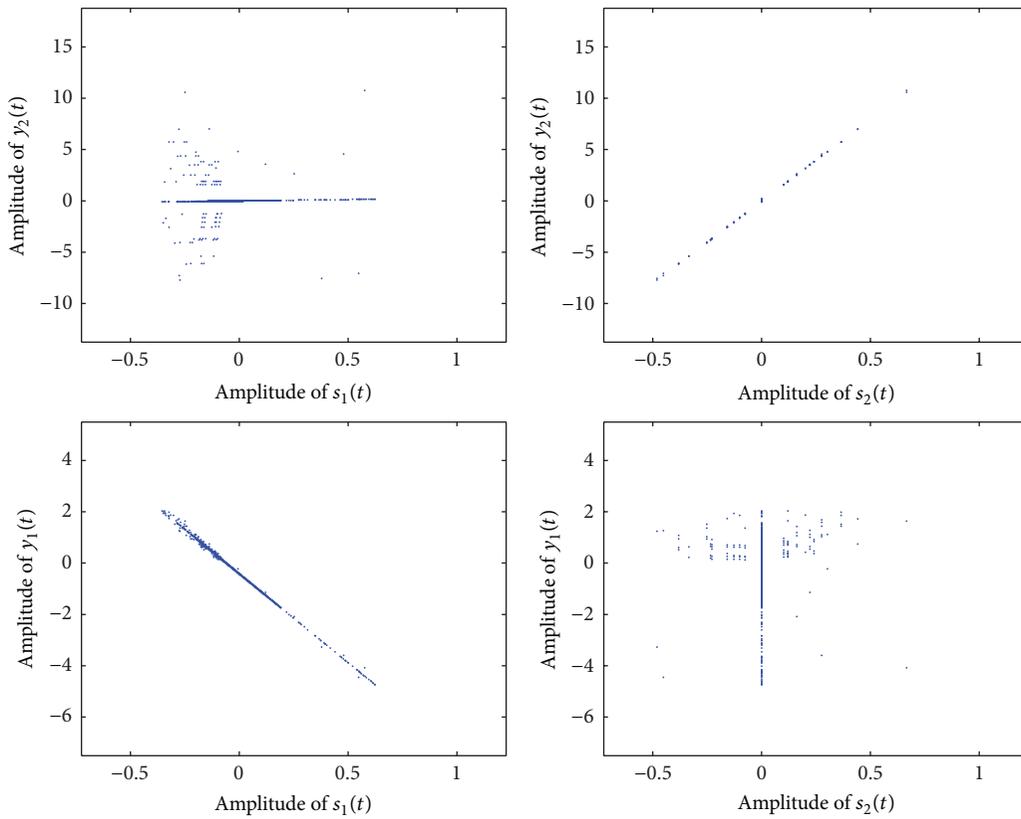


FIGURE 13: The composite scattering plot with FastICA.

In the experiment of extracting VLP signal, the comparison of similarity coefficient matrix between RFastICA and FastICA algorithm was shown in Table 2.

Extracted signal  $y_1(t)$  was the estimation of source signal  $s_1(t)$  and the phase was the same in Figure 12 and the opposite in Figure 13. Extracted signal  $y_2(t)$  was the estimation of source signal  $s_2(t)$  and the phase was the same in Figures 12 and 13.

From the above experiments, the performance of R-FastICA is superior to the FastICA obviously with higher similarity coefficient and high separation precision.

## 5. Conclusion

In this paper, R-FastICA algorithm and FastICA algorithm were adapted to extract random signals and to separate VLP signal from ECG signal. We believed that our study produced two important results. Firstly, we proposed a new method through the combination of related function and negative entropy and separated independent components by maximizing new objective function in the experiments. On the other hand, the experiments showed that R-FastICA method outperformed traditional FastICA method with higher similarity coefficient and high separation precision.

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## Acknowledgments

This project is supported by The General Object of National Natural Science Foundation (no. 61371062), Youth Science Foundation Project of National Natural Science Foundation (no. 61303207), Ministry of Education in 2012 Colleges and Universities by the Specialized Research Fund for the Doctoral Program of Joint Funding Subject (no. 20121402120020), Shanxi Province Science and Technology Development Project, Industrial Parts (no. 20120321024-01), Shanxi International Science and Technology Cooperation Project (no. 2012081031), Science and Technology Activities Project of Study Abroad Returnees in Shanxi Province in 2012 (Funded by Shanxi province human resources and social security hall), and Research Project supported by Shanxi scholarship council of China (no. 2013-032).

## References

- [1] C. J. James and C. W. Hesse, "Independent component analysis for biomedical signals," *Physiological Measurement*, vol. 26, no. 1, pp. R15–R39, 2005.
- [2] Y. Feng, M.-Y. He, J.-H. Song, and J. Wei, "ICA-based dimensionality reduction and compression of hyperspectral images," *Journal of Electronics & Information Technology*, vol. 29, no. 12, pp. 2871–2875, 2007.
- [3] D.-W. Wang, H. Ji, and Y.-J. Wang, "Feature-level fusion recognition based on complex-valued independent component analysis," *Optics and Precision Engineering*, vol. 17, no. 8, pp. 2024–2031, 2009.
- [4] X. Huang and H. Zeng, "Efficient variant of FastICA algorithm for speech features extraction," *Lecture Notes in Electrical Engineering*, vol. 113, pp. 929–935, 2012.
- [5] H. Mehrabian, L. Lindvere, B. Stefanovic, and A. L. Martel, "A constrained independent component analysis technique for artery-vein separation of two-photon laser scanning microscopy images of the cerebral microvasculature," *Medical Image Analysis*, vol. 16, no. 1, pp. 239–251, 2012.
- [6] A. J. Bell and T. J. Sejnowski, "An information-maximization approach to blind separation and blind deconvolution," *Neural Computation*, vol. 7, no. 6, pp. 1129–1159, 1995.
- [7] X.-M. Yang, X.-D. Zheng, and X.-H. Liu, "The clinical significance of ventricular late potentials," *Journal of Clinical Research*, vol. 25, no. 11, pp. 1123–1124, 2012.
- [8] A. S. Zandi and M. H. Moradi, "High-resolution ECG analysis: a fuzzy approach to detect ventricular late potentials using a wavelet-based vector magnitude waveform," in *Proceedings of the IEEE Workshop on Signal Processing Systems—Design and Implementation*, pp. 446–450, IEEE Press, November 2005.
- [9] P. Bonato, R. Bettini, G. Speranza, F. Furlanello, and R. Antolini, "Improved late potential analysis in frequency," *Medical Engineering and Physics*, vol. 17, no. 3, pp. 232–238, 1995.
- [10] F. Castells, J. Igual, J. J. Rieta, C. Sanchez, and J. Millet, "Atrial fibrillation analysis based on ICA including statistical and temporal source information," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 93–96, IEEE Press, April 2003.
- [11] H.-X. Zhang and L.-X. Jia, "The detection of ventricular late potential signal based on BSS," *Journal of Jishou University (Natural Science Edition)*, vol. 29, no. 1, pp. 62–65, 2008.
- [12] Y. Wang, *The identification of ventricular late potentials based on wavelet transform and independent component analysis [M.S. thesis]*, Chongqing University, 2007.
- [13] K. She, H.-M. Pu, F.-W. Zheng, and M.-T. Zhou, "Adaptive independent component analysis under multisensing," *Journal of the University of Electronic Science and Technology of China*, vol. 36, no. 1, pp. 11–13, 2007.
- [14] Y. Song and D. Li, "Analysis method and search algorithm to ICA," *Journal of System Simulation*, vol. 21, no. 8, pp. 2346–2349, 2009.
- [15] X.-X. Zhang and T.-S. Qiu, "Random variable analogy based ICA method," *Journal of Electronics & Information Technology*, vol. 31, no. 9, pp. 2104–2108, 2009.
- [16] F. R. Bach and M. I. Jordan, "Kernel independent component analysis," *Journal of Machine Learning Research*, vol. 3, no. 1, pp. 1–48, 2002.
- [17] A. Hyvärinen and E. Oja, "Independent component analysis: algorithms and applications," *Neural Networks*, vol. 13, no. 4–5, pp. 411–430, 2000.
- [18] Y.-Q. Du and H. Huang, "Fetal electrocardiogram extraction based on independent component and wavelet analyses," *Journal of Clinical Rehabilitative Tissue Engineering Research*, vol. 15, no. 13, pp. 2394–2397, 2011.
- [19] J. Wan, X.-H. Zhang, J.-H. Rao, and Q.-P. Hu, "An independent component analysis algorithm using signal noise ratio," *Control and Decision*, vol. 27, no. 9, pp. 1376–1380, 2012.
- [20] C. Vaya, J. J. Rieta, C. Sanchez, and D. Moratal, "Convolutional blind source separation algorithms applied to the electrocardiogram of atrial fibrillation: study of performance," *IEEE*

*Transactions on Biomedical Engineering*, vol. 54, no. 8, pp. 1530–1533, 2007.

- [21] Z.-H. Zhi, X.-Y. Zhang, and H.-R. Jia, “Flexible independent component analysis based on generalized gamma distribution,” *Journal of Taiyuan University of Technology*, vol. 40, no. 1, pp. 4–7, 2009.
- [22] R. Sivakumar, B. Muralikumar, B. Saranya, and V. P. Shanthiprabha, “Analysis of ventricular late potentials using wavelets and FFT spectrum,” *International Journal of Biomedical Engineering and Technology*, vol. 6, no. 1, pp. 76–92, 2011.

## Research Article

# Towards a Unified Sentiment Lexicon Based on Graphics Processing Units

Liliana Ibeth Barbosa-Santillán<sup>1</sup> and Inmaculada Álvarez-de-Mon y-Rego<sup>2</sup>

<sup>1</sup> Department of Computer Science, Universidad de Guadalajara, Periferico Norte 799, Modulo L-308, Los Belenes, 45100 Guadalajara, JAL, Mexico

<sup>2</sup> Lingüística Aplicada a la Ciencia y la Tecnología, Universidad Politécnica de Madrid, Madrid, Spain

Correspondence should be addressed to Liliana Ibeth Barbosa-Santillán; [lbarbosa@alumnos.fi.upm.es](mailto:lbarbosa@alumnos.fi.upm.es)

Received 16 July 2013; Accepted 11 October 2013; Published 13 March 2014

Academic Editor: Yudong Zhang

Copyright © 2014 L. I. Barbosa-Santillán and I. Álvarez-de-Mon y-Rego. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper presents an approach to create what we have called a Unified Sentiment Lexicon (USL). This approach aims at aligning, unifying, and expanding the set of sentiment lexicons which are available on the web in order to increase their robustness of coverage. One problem related to the task of the automatic unification of different scores of sentiment lexicons is that there are multiple lexical entries for which the classification of positive, negative, or neutral  $\{P, N, Z\}$  depends on the unit of measurement used in the annotation methodology of the source sentiment lexicon. Our USL approach computes the unified strength of polarity of each lexical entry based on the Pearson correlation coefficient which measures how correlated lexical entries are with a value between 1 and  $-1$ , where 1 indicates that the lexical entries are perfectly correlated, 0 indicates no correlation, and  $-1$  means they are perfectly inversely correlated and so is the UnifiedMetrics procedure for CPU and GPU, respectively. Another problem is the high processing time required for computing all the lexical entries in the unification task. Thus, the USL approach computes a subset of lexical entries in each of the 1344 GPU cores and uses parallel processing in order to unify 155802 lexical entries. The results of the analysis conducted using the USL approach show that the USL has 95.430 lexical entries, out of which there are 35.201 considered to be positive, 22.029 negative, and 38.200 neutral. Finally, the runtime was 10 minutes for 95.430 lexical entries; this allows a reduction of the time computing for the UnifiedMetrics by 3 times.

## 1. Introduction

Written language has been the preferred medium of communication in order to express facts, assumptions, and opinions. The web has facilitated the connection of speakers overcoming the barriers imposed by location, language used, customs, context, culture, and so forth through electronic devices. Content producers are increasing their activity in blogs, web pages, portals, social networks, e-mails, chats, and so forth.

Surprisingly, the speaker boundaries of nationality are no longer distinguished, even using the Global Positioning System, due to the growth of a global migration and the merging of languages by multilingual professional communities.

In many countries, multilingual features occur in many families where parents have several mother tongues. They use

a common language at home as they live in a country where another language is spoken. The children swap between an average of four languages simultaneously excluding the extra languages that they learn in school. For instance, according to the United States Census in 2007 [1], the percentage of individuals around five years and older who speak only English at home is 80.3% and who use a language other than English at home is 19.7%, among them Spanish accounts for 62.3% followed by Asian and Pacific Island languages with 15.0%.

On the other hand, the global economy is based on the digital applications such as e-commerce and online entertainment, social media including social enterprise, digital media advertising, the Internet, and cloud computing. For that reason, knowing the opinion of citizens becomes essential because they are increasingly active in the content production

which enterprises, researchers, government, and intelligence services consider that can be monitored.

To analyse the web to discover sentiment is a daunting task due to the difficulty of getting accurate results. Nevertheless, machine learning algorithms have obtained good results classifying sentiment within specialized domains and using controlled corpora.

Most existing studies have been conducted using cluster or statistic analysis in order to classify sentences, paragraphs, and documents. Furthermore, several rates and user profiles have been used in the collaborative assessment of services.

In summary, a collective interest is to understand the thought of a global society. In this context, structured linguistic resources are vital and they should be supported by a group of linguists working on a global level.

Lexicons are atomic linguistic resources necessary for processing information automatically. The web and the information explosion is making the available lexicons insufficient because the web is heterogeneous, multilingual, and dynamic. Even though there are approximations for automatically creating sentiment lexicons, they definitely should be improved in the areas of verification task and expert assessment.

Our research focuses on the four languages with the greatest number of first-language speakers including Chinese with 1,197 million, Spanish with 406 million, English with 335 million, and Portuguese with 202 million (based on Lewis, et al. [2]). Each family contains 15,000,000, 350,000, 88,000, and X words, respectively, according to the Cambridge, RAE, Oxford, and Grande Dicionário Houaiss da Língua Portuguesa dictionaries in 2013. Expecting to produce a 100% robust sentiment lexicon in our research is a titanic task. However, to automatically increase their robustness with improved quality is possible.

Another important challenge is the representation of lexically encoded knowledge and the researchers are suggesting new ways to do this. Moreover, their structures are different from each other, making it difficult to reuse, so their resources become a problem of interoperability.

In addition, the same lexical entry can be found in many sources having distinct assessments. In this case, the unification task is key since one of our goals is to compare the strength of polarity between sources of information and their symbols in several languages. However, the problem of unifying the strength of polarity is primarily a problem of processing power due to the size of the sentiment lexicon, which makes a hand-by-hand analysis simply not feasible.

If we want to know the correlation of a lexical entry to the rest of the Unified Sentiment Lexicon, then the number of possibilities is  $9.08E + 009$ . So computing is a problem of time as the calculation involved is huge.

Lexical resources, especially those semantically annotated, are time consuming and require a lot of effort; thus, we tried to use as much already existing work as possible in our effort to build a Unified Sentiment Lexicon.

Sentiment Lexicons that our research used are

**SL1** SentiWordNet developed by Istituto di Scienza e Tecnologia dell'Informazione,

**SL2** Bing Liu Sentiment Lexicon developed by Illinois University,

**SL3** MPQA lexicon developed by Pittsburgh University,

**SL4** NTU Sentiment Dictionary (NTUSD) developed by the Institute of Information Science of Taiwan,

**SL5** PanAmerican sentiment lexicon developed by the Polytechnic University of Madrid,

**SL6** Spanish Travel Subjective Lexicon (STSL) developed by the Polytechnic University of Madrid.

Our research questions are

**Q1** is it possible to unify the sentiment lexicons available on the web and align and expand them automatically?

**Q2** Is it possible to transform a Unified Sentiment Lexicon into a generative lexicon based on a core ontology?

The following set of hypotheses covers the main features of the proposed solutions:

**H1** the unification of sentiment lexicons allows for a robust linguistic resource,

**H2** given different strengths of polarity of the same lexical entry, it is possible to compute a unified value,

**H3** unification calculus of each one of the lexical entries with GPUs' local and global memory allow the reduction of hard disk access and increase processing speed.

Compared with previous work, the major contributions of this paper are as follows:

**C1** a cluster of sentiment lexicons has been unified automatically and validated by experts,

**C2** the Unified Sentiment Lexicon has been expanded with two more sentiment lexicons that were developed by our research group Communication in Specialized Domains,

**C3** the task of unification uses parallel processing for computing each lexical entry with GPUs,

**C4** USL compute was accelerated by 3 times in data processing,

**C5** robustness of coverage of the Unified Sentiment Lexicon,

**C6** a uniform representation of lexically encoded knowledge.

In summary, this paper describes the Unified Sentiment Lexicon (USL) approach for aligning, unifying, and expanding the sentiment lexicons available in an automatic way in order to increase their robustness of coverage obtaining as a result a large-scale Unified Sentiment Lexicon based on GPUs.

The remainder of the paper is structured as follows. Section 2 briefly presents the background of our work. Section 3 describes the USL approach. Section 4 describes how our USL approach was implemented and the different subtasks of the algorithm in detail. Section 5 presents details of our data sets, evaluation metrics, and the result. Finally, Section 6 presents our conclusions and future research.

## 2. Related Work

The related work considers the following: Section 2.1 data structures such as lexicons, specialized lexicons, and ontologies; Section 2.2 the methodologies available for building lexicons; and Section 2.3 the techniques of data processing focused on parallel processing and the kind of memory used for the TESLA architecture. Finally, we will present a summary table with the main features of the sentiment lexicons that are part of our study.

*2.1. Data Structures.* The lexical representation is founded in several data structures that form the basis of the linguistic resource, which is atomic. First, we will examine the lexicons; second, the generative lexicons; and finally, the ontologies.

*2.1.1. Lexicons.* According to Greame et al., a lexicon “is a list of words in a language along with some knowledge of how each word is used.” A lexicon can have monolingual or multilingual properties. Lexicons are either created manually [3–5], semiautomatic [6, 7], or automatically [8–11]. When the lexicon is built manually a group of experts can annotate all the words in a specific corpus; the assessment is performed by consensus and each lexical entry is checked in order to achieve excellence.

On the other hand, automatic lexicons can be produced, based on a specific corpus, where the lexical entries included far exceed the total number that can be compiled manually. However, to assess their quality is not an easy task.

In state of the art research [3, 4, 8–10], each group examines a collection of documents and produces their own lexicon. As a result, we have a number of lexicons—some of them are available on the web and others are not—we describe those in Section 2.1.3. In fact, we believe that all work carried out by universities and research groups should be used in a homogeneous way. Furthermore, these lexicons available should be reused using algorithms that facilitate data processing.

The generative lexicon was introduced by Pustejovsky [12, 13] in 1995 with the aim of encoding selectional knowledge in language. Differently to a generative lexicon, an enumerative lexicon only includes the different senses for each lexical entry. In Pustejovsky’s approach, there are four elements to encoding: lexical typing structure, argument structure, event structure, and qualia structure. For that reason Pustejovsky’s model deals with (a) the knowledge representation of the lexicon and the relation between an object and its constituents, (b) the formal role that distinguishes the object within a larger domain, (c) the purpose and function of the object, and (d) the factors involved in the origin of an object; all these constitute the qualia structure.

Bergler [14] said that there are significant efforts involved in building and sharing a big generative lexicon that will be a standard in the scientific community.

*2.1.2. Ontology.* According to Gruber [15] “an ontology is an explicit specification of a conceptualization.” In this sense, Graeme Hirst [16] said that “an ontology, as a nonlinguistic object that more directly represents the world, may provide an interpretation or grounding of word senses.”

The following supported sentiment ontologies are available: Ontology-Supported Polarity Mining [17] introduced in 2005, it was based on an ontology for movie reviews, with the positive or negative polarity determined from a collection of texts and the Chinese Emotion Ontology based on HowNet [18] which contains just under 5,500 verb concepts covering 113 different emotion categories.

*2.1.3. Sentiment Lexicons.* Our research has focused on four sentiment lexicons that are available on the web: the National Taiwan University Sentiment Dictionary (NTUSD), SentiWordNet, Bing Liu Sentiment Lexicon, and the Subjectivity Lexicon of Theresa Wilson et al. (MPQA). These are explained below.

The National Taiwan University Sentiment Dictionary (NTUSD) [19] was developed by Lun-Wei Ku et al. It is based on the Chinese Network Sentiment Dictionary and the web. There are 11,088 terms that qualify for the simplified version, of which 2,812 are positive and 8,276 are negative. The reason for having the NTUSD was to identify positive and negative sentiment words and their weights in a corpus of blogs and news on the basis of Chinese word structures. Lun-Wei Ku et al. suggested a method for annotating 192 documents in order to tag them as positive, negative, or neutral on three levels: words, sentences, and documents. The results were that the F-measure was 73.18% and 63.75% for verbs and nouns, respectively. When the sentiment words were mined together with topical words, they achieved an F-measure of 62.16% at the sentence level and 74.37% at the document level.

SentiWordNet [20] is a lexical resource produced by Istituto di Scienza e Tecnologie dell’Informazione, Italy. The main objective is to automatically estimate the value of all entries of WordNet [21] as positive, negative, or neutral assigning to each a weight between zero and one according of the value. The reason to create the SentiWordNet automatically was that the WordNet has more than 155,287 entries and annotating them manually would be a time consuming task. The result is a sentiment lexicon with 117,659 terms classified into the same four lexical categories as WordNet: adjectives, nouns, verbs, and adverbs.

Bing Liu Sentiment Lexicon [22, 23] is a lexicon where a small list of adjectives was manually created and tagged with positive or negative labels. It is domain-independent and he proposed a technique to grow this list using WordNet. He used a web-mining method to obtain a set of adjectives in the same way that the speaker wrote them. Thus, their lexicon has entries that are not in the English dictionary. The results obtained are 2,006 positive words and 4,764 negative words.

The Subjectivity Lexicon of Theresa Wilson et al. [3] is a lexical resource where the words that are subjective in most

contexts are marked as being strongly subjective and those that may only have subjective usages in certain contexts are marked as weakly subjective. The process of building it was manual. The words were also classified according to their categories as nouns, verbs, adjectives, and adverbs. The results are 8,221 clues (as she call them) where 4,913 are negative, 2,721 are positive, and 571 are neutral.

The Spanish Travel Subjective Lexicon (STSL) [4] was built *ad hoc* based on a web-based corpus of blogs that were analysed within the framework of appraisal theory [24]. The blogs were analysed to create a subcorpus of sentences annotated according to appraisal and these sentences were classified as positive or negative considering some contextual rules that could influence the strength of the polarity. These sentences were used to build the lexicon. The words were classified according to their categories as nouns, verbs, adjectives, and adverbs; multiword units were also included. The result was 1,610 terms of which 857 are positive and 753 are negative.

The PanAmerican Sentiment Lexicon approach aims to classify according to polarity a set of internet resources focused on an event. The approach is based on four components: a crawler, a filter, a synthesizer, and a polarity analyzer. The main function of the crawler component is to search and find data from internet resources related to the event of interest. After locating the data, the filter component processes the data in order to remove noise. The filter component only debugs internet resources that are associated with the event. At this point, the corpus consists of large posts containing large amounts of data from many countries and in many languages. The synthesizer component represents the amount of data into clusters with similar expressions using unsupervised learning. Finally, the polarity analyzer component classifies each lexical entry as positive, neutral, or negative. The lexical categories are noun, adjective, verb, adverb. Finally, the result arrived at of 6083 positive, 5300 negative, and 5000 neutral.

**2.2. Methodology.** We have identified several kinds of methodology for building sentiment lexicons and have classified them as follows: automatic, semiautomatic, and manual.

(1) *Automatic Methodology.* First, the crawler is used to obtain a set of lexical terms in a controlled domain. Next, data preprocessing is performed and terms are assessed and classified as positive, negative, or neutral. The evaluation task involves using a subset of annotated lexical entries created manually by experts in order to measure the accuracy of the results. However, one of the limitations is the quality of the results because undertaking the evaluation task manually is not feasible. One of the advantages of this methodology is the higher number of terms produced compared to the results of the manual methodology.

(2) *Semiautomatic Methodology.* When linguistic resources use this methodology both manual and automatic annotations are used. An initial lexicon is annotated manually and this subset is used for training the algorithm, which will predict the level of matching for automatically classifying each new lexical category in the lexicon.

(3) *Manual Methodology.* Experts manually annotate each lexical entry in the appropriate lexical category. The lexicon quality is high, but the depth of the lexicon is less than that obtained with other methodologies.

**2.3. Parallel Processing.** Parallel processing [25] allows the running of several jobs at the same time and to accelerate the process by producing answers concurrently. Graphics Processing Units (GPUs) [26] allow the implementation of several algorithms [27].

These algorithms have been proved to provide acceleration from  $2x$  to  $nx$  for specific problems [28].

Previous research has focused on image systems, simulations of fluids and molecular simulations with GPUS. Image systems [29] such as TechniScan uses ultrasonic waves to image the patient's chest in 20 minutes. Some other examples of the use of this technology are the following. The University of Cambridge was able to accelerate computational simulations of fluid dynamics [30] in order to perform rapid experimentation. Temple University performs molecular simulations [31] in order to reduce the environmental impact of detergents and cleaning agents. The time these simulation lasted was reduced from several weeks to a few hours.

We explored the use of GPU technology [32] in order to accelerate our data preprocessing.

TESLA [33] is a GPU architecture produced in 2003 by NVIDIA. It consists of a shared memory, constant cache, register file, double precision, Special Function Unit (SFU), Streaming Processor (SP), and a Warp Scheduler.

Memory is an essential component of high-performance computing. CUDA uses several types of memory [34] depending on the problem. The host memory is in the GPU system. CUDA provides APIs that enable faster access to the host memory by using the pager block and mapping the address direction on the GPU.

The memory device is in the GPU. It can be accessed by the dedicated memory controller. Data must be explicitly copied between the host memory and device memory. This memory can be organized and accessed in different ways [35].

Global memory can be static or dynamic. Access is by CUDA core pointers. Its main function is to translate the addresses.

Constant memory is read only. Its access is through a hierarchical cache optimized for transmission to several threads.

Local memory is stored in the stack: local variables which cannot be stored in records, parameters, and the return addresses of subroutines.

Texture memory is accessed by instructions for loading and storing. As well as constant memory, an independent cache is used in order to execute read only operations.

The GPU CUDA device is a multicore coprocessor. It is possible to log in through all the device memory without constraints. However, there will be variations in runtime according to the type of target memory.

TABLE 1: Sentiment lexicons that are available on the web and supported by universities.

Name	University	Positive	Negative	Neutral	Language	Methodology	Category	Order
Bing Liu	Illinois	2006	4783	0	English	Automatic	No	Alphabetic
MPQA	Pittsburg	2721	4913	571	English	Manual	N, V, Ad, Adv	Alphabetic
NTUSD	SINICA	2812	8276	0	Chinese	Semiautomatic	No	Phonetic
Pan American	UPM	6083	5300	5000	English Spanish Portuguese	Manual	N, V, Ad, Ad	Alphabetic
SentiWordNet	ISTI	857	753	0	English	Automatic	N, V, Ad, Ad N, V, Ad, Ad	No
STSL	UPM	19619	29792	89135	Spanish	Manual	Interjections diminutives phrases	Alphabetic

2.4. *Summary Table.* Table 1 summarizes the sentiment lexicons of our study according to several features such as name of the university where it was developed; depth of the lexical entries, which is the sum of all the positive, negative, or neutral entries; coverage of language; type of elaboration methodology; lexical categories; and ordering procedure.

### 3. The Approach Proposed

Our approach aims at aligning, unifying, and expanding the set of sentiment lexicons which are available on the web in order to increase their robustness of coverage. It is composed of ten components: FocusCrawlerEngine, SelectorLanguages, MetricSearcher, MetricTransformer, SentimentLexiconIntersection, LexicalEntries Substracter, LexicalEntriesDivisor, UnifiedMetrics, UnionSentimentLexiconEngine, and Lexicon2OntologyConverter.

The USL approach has as an input of sentiment lexicons which are supported by universities and which are available on the web. First, the SelectorLanguages component creates a group of sentiment lexicons according to their language and stores them in different knowledge bases. The MetricSearcher component performs an inspection of each one of the elements of the sentiment lexicons in order to identify if they have associated metrics. Then it saves the results in two knowledge bases: (a) MetricsLexicon and (b) NoMetricsLexicon. Next, the MetricTransformer component verifies if the metrics are not numerical in order to transform them with real values based on the original assessment. Consequently, our USL approach performs the intersection between all the sentiment lexicons. The common lexical entries are extracted with word, the strength of polarity, and sentiment lexicon values. Two knowledge bases are obtained as partial result: IntersectionLexicalEntries and NoIntersectionLexicalEntries.

The MetricsLexicon Knowledge base is the input for the LexicalEntries Substracter whose main function is to exclude all the IntersectionLexicalEntries. The USL approach is able to calculate a unified strength of polarity between all of the lexical entries of several sentiment lexicons. Thus, calculating the unified strength of polarity demands a high processing time because of the number of lexical entries. The LexicalEntriesDivisor split jobs in order to calculate the unification strength of polarity into balanced loads. The coprocessors

compute the degree of unified subjective for each lexical entry. Its calculation is based on a previous assessment of the sentiment's lexical sources and the incomplete information. Each coprocessor produces a lexical knowledge base with the score of the unified metric. The UnionSentimentLexiconEngine unifies all the knowledge bases into one. As a result we have the Unified Sentiment Lexicon (USL). Finally, the Lexicon2OntologyConverter performs a transformation from data to Ontology Web Language (OWL).

The ten components of USL approach will be described in detail.

3.1. *FocusCrawlerEngine.* Their main function is to find the sentiment lexicons available on the web that would be supported by universities. Then we can define the following: (1) MPQA Lexicon, (2) Bing Liu Sentiment Lexicon, (3) SentiWordNet, (4) NTU Sentiment Dictionary, (5) Spanish Travel Subjective Lexicon, and (6) PanAmerican Games Sentiment Lexicon.

Consider

$$MPQALexicon : \doteq \{Type, Length, Word, Position, Stemed, PriorPolarity\}, \quad (1)$$

where

$Type = \{t \mid t \text{ is a string which measures the subjectivity degree of each clue}\},$

$Length = \{l \mid l \text{ is a integer number that indicate the length of the clues}\},$

$Word = \{w \mid w \text{ is a string with the token or stem of the clue}\},$

$Position = \{p \mid p \text{ is a natural number than identify a lexical}\},$

$Stemed = \{s \mid s \text{ is a boolean than should match all unstemmed variants}\},$

$PriorPolarity = \{pp \mid pp \text{ is a string with the prior polarity of clue}\}.$

Consider

$$BingLiuSentimentLexicon : \doteq \{Words\}, \quad (2)$$

where  $Words = \{w \mid w \text{ is a string with a word that is positive or negative}\}.$

Consider

$$Travel : \doteq \{NP, NN, AP, AN, VP, VN, adP, AdN\}, \quad (3)$$

where

$$\begin{aligned} NP &= \{np \mid np \text{ is a string with a word that is noun and positive}\}, \\ NN &= \{nn \mid nn \text{ is a string with a word that is noun and negative}\}, \\ AP &= \{ap \mid ap \text{ is a string with a word that is adjective and positive}\}, \\ AN &= \{an \mid an \text{ is a string with a word that is adjective and positive}\}, \\ VP &= \{vp \mid vp \text{ is a string with a word that is verb and positive}\}, \\ VN &= \{vn \mid vn \text{ is a string with a word that is verb and positive}\}, \\ AdP &= \{adp \mid adp \text{ is a string with a word that is adverb and positive}\}, \\ AdN &= \{and \mid and \text{ is a string with a word that is adverb and positive}\}. \end{aligned}$$

Consider

$$NTUSD : \doteq \{Words\}, \quad (4)$$

where  $Words = \{w \mid w \text{ is a string with a word that is positive or negative}\}$ .

Consider

$$SentiWordNet : \doteq \{POS, ID, PosScore, NegScore, SynsetTerms\}, \quad (5)$$

where

$$\begin{aligned} POS &= \{p \mid p \text{ is a character of WordNet}\}, \\ ID &= \{id \mid id \text{ is a character of WordNet}\}, \\ PosScore &= \{ps \mid ps \text{ is a real number with the positive score assigned by SentiWordNet}\}, \\ NegScore &= \{ns \mid ns \text{ is a real number with the negative score assigned by SentiWordNet}\}, \\ SynsetTerms &= \{st \mid st \text{ is a string number with the terms}\}. \end{aligned}$$

Consider

$$PanAmericanSL : \doteq \begin{cases} ID, Timestamp, Word, Postive, \\ Negative, Neutral, \\ Noun, Adjective, Verb, \\ Adverb, Language, \end{cases} \quad (6)$$

where

$$\begin{aligned} ID &= \{id \mid id \text{ is an integer number to identify a word}\}, \\ TimeStamp &= \{ts \mid ts \text{ is a date of assessment}\}, \end{aligned}$$

$$Word = \{w \mid w \text{ is a string with the word}\},$$

$$Positive = \{p \mid p \text{ is a boolean with 1 if it is positive}\},$$

$$Negative = \{n \mid n \text{ is a boolean with 1 if it is negative}\},$$

$$Neutral = \{nt \mid nt \text{ is a boolean with 1 if it is neutral}\},$$

$$Noun = \{noun \mid noun \text{ is a boolean with 1 if it is noun}\},$$

$$Adjective = \{ad \mid ad \text{ is a boolean with 1 if it is adjective}\},$$

$$Verb = \{v \mid v \text{ is a boolean with 1 if it is verb}\},$$

$$Adverb = \{ad \mid ad \text{ is a boolean with 1 if it is adverb}\},$$

$$Language = \{l \mid l \text{ is a string with the name of the language}\}.$$

3.2. *SelectorLanguages*. This identifies the language in a subset of lexical entries in order to search for specific words in four languages: Chinese, Spanish, English, and Portuguese. The result is the cluster of sentiment lexicons arranged by language, as shown in (1).

Consider

$$LanguageWords$$

$$\subseteq LexicalEntries \longleftrightarrow (\forall x) (x \in LanguageWords$$

$$\mapsto x \in LexicalEntries). \quad (7)$$

3.3. *MetricSearcher*. It is responsible for selecting the strength of polarity label of each of the sentiment lexicons clusters. For example, ‘‘PriorPolarity,’’ ‘‘PosScore,’’ ‘‘NegScore,’’ and ‘‘ScoreSubjectivity,’’ among others. Besides, it searches for strength of polarity and indicates whether the values are numerical or nominal. It splits its result in two knowledge bases: MetricsLexicon and NoMetricsLexicon.

Consider

$$MetricsLexicon = \{x \mid (x \in LexicalEntries)$$

$$\wedge (x \notin NoMetrics)\},$$

$$NoMetricsLexicon = \{y \mid (y \in LexicalEntries) \quad (8)$$

$$\wedge (y \notin Metrics)\}.$$

3.4. *MetricTransformer*. The MetricTransformer works by transforming the strength of polarity nominal value into the real value of each sentiment lexicon. It has two variables: type and pos. Type can take two values: *strongsub* = .9 and *weaksbj* = .5. Pos can take four values: *adj* = .9, *verb* = 1, *adverb* = .8, and *noun* = .7. The new strength of polarity is the multiplication between the two variables.

Consider

$$NewStrengthPolarity = type * pos. \quad (9)$$

3.5. *SentimentLexiconIntersection*. This component compiles with the intersection for all the lexical entries of each sentiment lexicon cluster. It aims to identify which lexical entries appear more than once in order to select them for processing.

Therefore, the two knowledge bases are IntersectionLexicalEntries and NoIntersectionalLexicalEntries. For example, the cluster of sentiment lexicons grouped by English language has four elements  $EnglishCluster = \{PanAmericanSentimentLexicon, BingLiuSentimentLexicon, SentiWordNet, and MPQALexicon\}$ . These intersections are shown in (10).

Consider

$$\begin{aligned}
SentiWordNet \cap BingLiu &= \{x \mid (x \in SentiWordNet) \\
&\quad \wedge (x \in BingLiu)\}, \\
SentiWordNet \cap MPQA &= \{x \mid (x \in SentiWordNet) \\
&\quad \wedge (x \in MPQA)\}, \\
MPQA \cap BingLiu &= \{x \mid (x \in MPQA) \wedge (x \in BingLiu)\}, \\
SentiWordNet \cap BingLiu \cap MPQA &= \{x \mid (x \in SentiWordNet) \wedge (x \in BingLiu) \\
&\quad \wedge (x \in MPQA)\}, \\
PanAmerican \cap SentiWordNet &= \{x \mid (x \in PanAmerican) \wedge (x \in SentiWordNet)\}, \\
BingLiu \cap PanAmerican &= \{x \mid (x \in BingLiu) \wedge (x \in PanAmerican)\}, \\
PanAmerican \cap SentiWordNet \cap BingLiu &= \{x \mid (x \in PanAmerican) \\
&\quad \wedge (x \in SentiWordNet) \wedge (x \in BingLiu)\}, \\
PanAmerican \cap MPQA &= \{x \mid (x \in PanAmerican) \wedge (x \in MPQA)\}, \\
SentiWordNet \cap PanAmerican \cap MPQA &= \{x \mid (x \in SentiWordNet) \wedge (x \in PanAmerican) \\
&\quad \wedge (x \in MPQA)\}, \\
PanAmerican \cap BingLiu \cap SentiWordNet \cap MPQA &= \{x \mid (x \in PanAmerican) \wedge (x \in BingLiu) \\
&\quad \wedge (x \in SentiWordNet) \wedge (x \in MPQA)\}.
\end{aligned} \tag{10}$$

3.6. *LexicalEntriesSubstracter*. This gets the rest of all the elements that have been assessed by each university. It subtracts MetricsLexicon from IntersectionLexicalEntries.

Consider

$$\begin{aligned}
NoIntersectionLexicon &= InitialLexicon(x) - IntersectionLexicon, \\
NoIntersectionalLexicon &= def \{s \mid s \in InitialLexicon, \\
&\quad s \notin IntersectionLexicon\}.
\end{aligned} \tag{11}$$

3.7. *LexicalEntriesDivisor*. It has as its input the intersection of all the lexical entries. It divides the knowledge base into equal parts for processing.

Consider

$$N = \frac{LexicalEntriesTotal}{NumberCores}. \tag{12}$$

3.8. *UnifiedMetrics*. This performs an estimate of each lexical entry of the IntersectionLexicon in order to predict its value. There are two procedures: (1)  $UnifiedMetrics_{CPU}$  and (2)  $UnifiedMetrics_{GPU}$ .

$UnifiedMetrics_{CPU}$  uses a Pearson correlation formula as shown in (13) applied between the Unified Sentiment Lexicon and each of the sentiment Lexicons by cluster.

$UnifiedMetrics_{GPU}$  algorithm in detail is explained in Section 4.

Consider

$$r = \frac{\sum XY - \sum X \sum Y / N}{\sqrt{(\sum x^2 - (\sum x)^2 / N) (\sum y^2 - (\sum y)^2 / N)}}. \tag{13}$$

3.9. *UnionSentimentLexiconEngine*. Its function is to join all the result knowledge bases of the coprocessors together and as output the Unified Sentiment Lexicon is obtained.

Consider

$$(\forall x) (\exists y) (\forall z) [z \in y \longleftrightarrow (\exists t) (t \in x \wedge z \in t)], \tag{14}$$

where  $UnifiedSentimentLexicon_1 \cup \dots \cup \dots UnifiedSentimentLexicon_n \leftarrow x \in this \Leftrightarrow x \in UnifiedSentimentLexicon_1$  or  $x \in UnifiedSentimentLexicon_n$ .

3.10. *Lexicon2OntologyConverter*. Their main function is to transform the Unified Sentiment Lexicon into a Domain Ontology: OntoLexicon as defined as follows.

3.10.1. *OntoLexicon*. The OntoLexicon Ontology is a conceptual description based on a lexicon of the subjective words in Natural Language as shown in (15). The OntoLexicon Ontology consists of four disjoint sets  $C, R, A,$  and  $\tau$ , where  $C$  means concept identifiers (16),  $R$  means relation identifiers (17) and (18),  $A$  means attribute identifiers (19), and  $\tau$  means data types (20).

Consider

$$OntoLexicon := (C, \leq c, R, \gamma_R, \leq_R, A, \gamma_A, \tau). \tag{15}$$

The set  $C$  of concepts is

$$C := \begin{cases} Adjectives, NegativeAdjectives, \\ PositiveAdjectives, Adverbs, NegativeAdverbs, \\ PositiveAdverbs, Verbs, NegativeVerbs, \\ PositiveVerbs, Nouns, NegativeNouns, \\ PositiveNouns. \end{cases} \tag{16}$$

The set  $R$  of relations is

$$R := \begin{cases} entry\_of, document\_of, paragraph\_of, \\ sentence\_of, adverb\_in, articles\_in, \\ prepositions\_in, nouns\_in, adjectives\_in, \\ verbs\_in, subject\_of, \\ predicate\_of, \end{cases} \tag{17}$$

where the relation hierarchy defines that Lexical has the relation *entry\_of* that belongs to SentimentLexicon. Corpora has the relation *document\_of* of that belongs to documents, following the same logic where the rest of the relations are defined as

$$\begin{aligned}
\gamma R(\text{entry\_of}) &= (\text{Lexical}, \text{SentimentLexicon}), \\
\gamma R(\text{document\_of}) &= (\text{Documents}, \text{Corpora}), \\
\gamma R(\text{paragraph\_of}) &= (\text{Paragraphs}, \text{Documents}), \\
\gamma R(\text{sentence\_of}) &= (\text{Sentences}, \text{Paragraphs}), \\
\gamma R(\text{adverbs\_in}) &= (\text{Adverbs}, \text{Sentences}), \\
\gamma R(\text{articles\_in}) &= (\text{Articles}, \text{Sentences}), \\
\gamma R(\text{prepositions\_in}) &= (\text{Prepositions}, \text{Sentences}), \\
\gamma R(\text{nouns\_in}) &= (\text{Nouns}, \text{Sentences}), \\
\gamma R(\text{adjectives\_in}) &= (\text{Adjectives}, \text{Sentences}), \\
\gamma R(\text{verbs\_in}) &= (\text{Verbs}, \text{Sentences}), \\
\gamma R(\text{subject\_in}) &= (\text{Subjects}, \text{Sentences}), \\
\gamma R(\text{predicate\_in}) &= (\text{Predicates}, \text{Sentences}).
\end{aligned} \tag{18}$$

The set  $A$  of attribute identifiers is

$$A := \begin{cases} \text{sentimentlexicon, author, strengthofpolarity,} \\ \text{paragraph, sentence, subject,} \\ \text{predicate, article, noun, nounN,} \\ \text{nounP, verb, verbN, verbP,} \\ \text{adverb, adverbN, adverbP, adjective,} \\ \text{adjectiveN, adjectiveP.} \end{cases} \tag{19}$$

The set  $\tau$  of datatypes contains only one element, a string, is shown

$$\tau := (\text{string}). \tag{20}$$

The first axiom defines the concept NegativeAdverbs as equivalent to saying that there is a negative adverb which stands in a *adverb\_in* relation with the corresponding sentence, following the same logic where the rest of the axioms are defined as

$$\begin{aligned}
\forall x (\text{NegativeNouns}(x) &\longleftrightarrow \exists y \wedge \text{noun\_in}(x, y) \\ &\quad \wedge \text{Sentences}(y)), \\
\forall x (\text{PositiveNouns}(x) &\longleftrightarrow \exists y \wedge \text{noun\_in}(x, y) \\ &\quad \wedge \text{Sentences}(y)), \\
\forall x (\text{NegativeAdjective}(x) &\longleftrightarrow \exists y \wedge \text{adjective\_in}(x, y) \\ &\quad \wedge \text{Sentences}(y)), \\
\forall x (\text{PositiveAdjective}(x) &\longleftrightarrow \exists y \wedge \text{adjective\_in}(x, y) \\ &\quad \wedge \text{Sentences}(y)), \\
\forall x (\text{NegativeAdverbs}(x) &\longleftrightarrow \exists y \wedge \text{adverb\_in}(x, y) \\ &\quad \wedge \text{Sentences}(y)), \\
\forall x (\text{PositiveAdverbs}(x) &\longleftrightarrow \exists y \wedge \text{adverb\_in}(x, y) \\ &\quad \wedge \text{Sentences}(y)), \\
\forall x (\text{NegativeVerbs}(x) &\longleftrightarrow \exists y \wedge \text{verb\_in}(x, y) \\ &\quad \wedge \text{Sentences}(y)),
\end{aligned}$$

$$\begin{aligned}
\forall x (\text{PositiveVerbs}(x) &\longleftrightarrow \exists y \wedge \text{verb\_in}(x, y) \\ &\quad \wedge \text{Sentences}(y)).
\end{aligned} \tag{21}$$

#### 4. Algorithm in Detail

The input of a Unified Sentiment Lexicon (USL) approach consists of the sentiment lexicons that are available on the web and supported by universities. The USL approach then processes all of them. The result is the Unified Sentiment Lexicon (USL). Here, we will describe the algorithm in detail.

The first step is to group the sentiment lexicons into clusters by language as following:

$$\begin{aligned}
x &= \{\text{ChineseLanguage}, \text{SpanishLanguage}, \\ &\quad \text{EnglishLanguage}, \text{PortugueseLanguage}\}, \\
\text{Cluster}_{x_1} &= \{\text{SentimentLexicon}_1, \dots, \\ &\quad \text{SentimentLexicon}_n\}.
\end{aligned} \tag{22}$$

The second step is to search lexical entries that have been assessed by each sentiment lexicon. In the assessment task, some authors and their methods have used nominal values, while others have used real values. If they are linguistic values, then the USL approach transforms them into real values. There must then be an intersection of lexical entries in at least two sentiment lexicons in order to unify the strength of polarity of several sentiment lexicons into one.

Following this, our approach calculates the Pearson correlation score between each sentiment lexicon and the USL by obtaining as many constants as there are sentiment lexicons in the cluster. For example, if the cluster belongs to the English language, then there are four constants that fall into each sentiment lexicon, as shown in the Pearson correlation set  $\text{PearsonCorrelation} = \{p_1, p_2, p_3, p_4\}$ . This calculation is performed only once and executed by the CPU.

Since the number of lexical entries is high, the computation of the USL score should be divided into several coprocessors (cores) in order to accelerate the process. In fact, each coprocessor of the GPU has as an input: (a) the strength of polarity of  $n$  lexical entries and (b) the vector with Pearson values. Each coprocessor computes the strength of polarity of every lexical entry until there are no lexical entries left. The score for each lexical entry is multiplied by the Pearson correlation between all the sentiment lexicons, as shown in (23) and Table 2.

Consider

$$\begin{aligned}
\alpha_i &= p_1 * v_i, \\
\beta_i &= p_2 * w_i, \\
\gamma_i &= p_3 * y_i, \\
\delta_i &= p_4 * z_i.
\end{aligned} \tag{23}$$

In addition, USL performs a total of subjectivity sums, as shown in (24) and Table 2.

Consider

$$\varepsilon_i = \alpha_i + \beta_i + \gamma_i + \delta_i. \tag{24}$$

TABLE 2: The process to calculate the USL strength of polarity.

Words	Sentiment Lexicon <sub>1</sub>	Sentiment Lexicon <sub>2</sub>	Sentiment Lexicon <sub>3</sub>	Sentiment Lexicon <sub>n</sub>	$\alpha$	$\beta$	$\gamma$	$\delta$	$\epsilon$	$\zeta$	USL
Word <sub>1</sub>	$v_1$	$x_1$	$y_1$	$X$	$\alpha_1$	$\beta_1$	$\gamma_1$	$\delta_1$	$\epsilon_1$	$\zeta_1$	$\mathbf{usl}_1 = \epsilon_1/\zeta_1$
Word <sub>2</sub>	$X$	$x_2$	$y_2$	$z_2$	$\alpha_2$	$\beta_2$	$\gamma_2$	$\delta_2$	$\epsilon_2$	$\zeta_2$	$\mathbf{usl}_2 = \epsilon_2/\zeta_2$
Word <sub>3</sub>	$X$	$x_3$	$y_3$	$z_3$	$\alpha_3$	$\beta_3$	$\gamma_3$	$\delta_3$	$\epsilon_3$	$\zeta_3$	$\mathbf{usl}_3 = \epsilon_3/\zeta_3$
Word <sub>4</sub>	$v_4$	$x_4$	$y_4$	$z_4$	$\alpha_4$	$\beta_4$	$\gamma_4$	$\delta_4$	$\epsilon_4$	$\zeta_4$	$\mathbf{usl}_4 = \epsilon_4/\zeta_4$
Word...	$v...$	$x...$	$X...$	$z...$	$\alpha...$	$\beta...$	$\gamma...$	$\delta...$	$\epsilon...$	$\zeta...$	$\mathbf{usl}... = \epsilon.../\zeta...$
Word <sub>n</sub>	$v_n$	$X$	$y_n$	$z_n$	$\alpha_n$	$\beta_n$	$\gamma_n$	$\delta_n$	$\epsilon_n$	$\zeta_n$	$\mathbf{usl}_n = \epsilon_n/\zeta_n$

```

(1) procedure UnifiedSentimentLexicon(seeds)
(2)   SentimentLexicons  $\leftarrow$  FocusCrawlerEngine(seeds);
(3)   Clusters  $\leftarrow$  SelectorLanguages(SentimentLexicons);
(4)   for  $i \leftarrow 1, \text{NumberOfClusters}$  do
(5)     for  $j \leftarrow 1, \text{NumberOfSentimentLexicons}$  do
(6)       for  $k \leftarrow 1, \text{NumberOfLexicalEntries}$  do
(7)         if MetricSearcher(LexicalEntry( $k$ )) = 1 then
(8)           MetricsLexicon( $j$ )( $k$ )  $\leftarrow$  LexicalEntry( $k$ );
(9)         else
(10)          NoMetricsLexicon( $j$ )( $k$ )  $\leftarrow$  LexicalEntry( $k$ );
(11)        end if
(12)        if MetricTransformer(NoMetricsLexicon( $j$ )( $k$ ))  $\geq 0$  then
(13)          MetricsLexicon( $j$ )( $k$ )  $\leftarrow$  NoMetricsLexicon( $j$ )( $k$ );
(14)        end if
(15)      end for
(16)    end for
(17)  end for
(18)  if SentimentLexiconIntersection(Clusters(MetricsLexicon)) = 1 then
(19)    IntersectionLexicalEntries( $j$ )( $k$ )  $\leftarrow$  Clusters(MetricsLexicon);
(20)  else
(21)    NoIntersectionLexicalEntries( $j$ )( $k$ )  $\leftarrow$  Clusters(MetricsLexicon);
(22)  end if
(23)  NoIntersectionLexicalEntries( $j$ )( $k$ )  $\leftarrow$  LexicalEntriesSubstracter
(24)    (SentimentLexicon, IntersectionLexicalEntries);
(25)   $n \leftarrow$  LexicalEntriesDivisor(IntersectionLexicalEntries, NumberOfCoresGPU);
(26)  for  $i \leftarrow 1, \text{NumberOfClusters}$  do
(27)    for  $j \leftarrow 1, n$  do
(28)       $r \leftarrow$  UnifiedMetricsCPU(Cluster( $i$ )(SentimentLexicons));
(29)      USL( $i$ )  $\leftarrow$  UnifiedMetricsGPU( $n, r, \text{Cluster}(i)(\text{IntersectionLexicalEntries})$ );
(30)      UnifiedSentimentLexicon  $\leftarrow$  UnionSentimentLexiconEngine( $j$ );
(31)    end for
(32)  end for
(33)  OntoLexicon  $\leftarrow$  Lexicon2OntologyConverter(UnifiedSentimentLexicon);
(34) end procedure

```

ALGORITHM 1: The main USL approach.

The USL score is normalized by dividing the total number of subjectivity for each lexical entry by the Pearson correlation sum of the lexical entries that were assessed  $\zeta_1 = p_1 + p_2 + p_3 + p_4$ , as follows:

$$USL_1 = \frac{\epsilon_1}{\zeta_1}. \quad (25)$$

The GPU results are the lexical entries combined with the USL score (these are input by the CPU). Their main function is to join all the partial results in the USL.

Finally, the CPU transforms the USL into an ontology called *OntoLexicon* in OWL language.

The pseudocode of the main USL approach is shown in Algorithm 1, and some of the procedures of the USL approach are shown in Algorithm 2.

## 5. Experimental Details and Performance Results

The following section includes a detailed description of how the experiment was conducted. The first part describes the

```

(1) procedure FocusCrawlerEngine(Seeds)
(2)   for  $i \leftarrow 1, \text{SizeSeeds}$  do
(3)     SentimentLexicons( $i$ )  $\leftarrow$  Download(Lexicon( $i$ ));
(4)   end for
(5) end procedure
(6) procedure SelectorLanguages(SentimentLexicons)
(7)   for  $i \leftarrow 1, \text{NumberOfSentimentLexicons}$  do
(8)     switch Language(SentimentLexicon( $i$ )) do
(9)       case English
(10)        assert(Cluster1  $\leftarrow$  SentimentLexicon( $i$ ))
(11)       case Spanish
(12)        assert(Cluster2  $\leftarrow$  SentimentLexicon( $i$ ));
(13)       case Chinese
(14)        assert(Cluster3  $\leftarrow$  SentimentLexicon( $i$ ));
(15)       case Portuguese
(16)        assert(Cluster4  $\leftarrow$  SentimentLexicon( $i$ ));
(17)     end for
(18) end procedure
(19) procedure MetricSearcher(LexicalEntry)
(20)   if LexicalEntry  $\geq 0$  then
(21)      $x \leftarrow 1$ ;
(22)   else
(23)      $x \leftarrow 0$ ;
(24)   end if
(25) end procedure
(26) procedure MetricTransformer(NoMetricsLexicon)
(27)   NoMetricsLexicon(newMetric)  $\leftarrow$  NoMetricsLexicon(type)*
      NoMetricsLexicon(pos);
(28) end procedure
(29) procedure SentimentLexiconIntersection(Clusters(MetricLexicon))
(30)   for  $i \leftarrow 1, \text{NumberOfCluster}$  do
(31)     for  $j \leftarrow 1, \text{NumberOfLexicalEntries}$  do
(32)       if MetricsLexicon(Cluster( $i$ )(LexicalEntry( $j$ )))
(33) = MetricsLexicon(Cluster( $i + 1$ )(LexicalEntry( $j$ ))) then
(34)         intersection  $\leftarrow 1$ ;
(35)       else
(36)         intersection  $\leftarrow 0$ ;
(37)       end if
(38)     end for
(39)   end for
(40) end procedure
(41) procedure LexicalEntriesSubstracter(SentimentLexicon, IntersectionLexicon)
(42)   LexiconSubstracter  $\leftarrow$  SentimentLexicon - IntersectionLexicon;
(43) end procedure
(44) procedure LexicalEntriesDivisor(LexicalEntriesTotal, NumberOfCores)
(45)   for  $i \leftarrow 1, \text{NumberOfCluster}$  do
(46)      $n \leftarrow \text{LexicalEntriesTotal} / \text{NumberOfCores}$ ;
(47)   end for
(48) end procedure

```

ALGORITHM 2: Some of the procedures of USL approach.

objectives of the experiment, and the second part focuses on the results obtained after the experiment was conducted.

The experimental setup had two objectives: to unify the sentiment lexicons and to carry out an analysis of our results by expert linguists.

The analysis was carried out in two ways: (1) where the USL approach ran automatically and (2) a linguistic evaluation of the quality of a subpart of the results obtained. The first

task was to obtain sentiment lexicons available on the web validated by universities using the FocusedCrawlerEngine and two sentiment lexicons developed by our research group. Communication in Specialized Domains were added.

The knowledge base of sentiment lexicons has been described in Section 3.1.

Figure 1 shows a partial view of the content of each of them: the Bing Liu sentiment lexicon appears in two files,



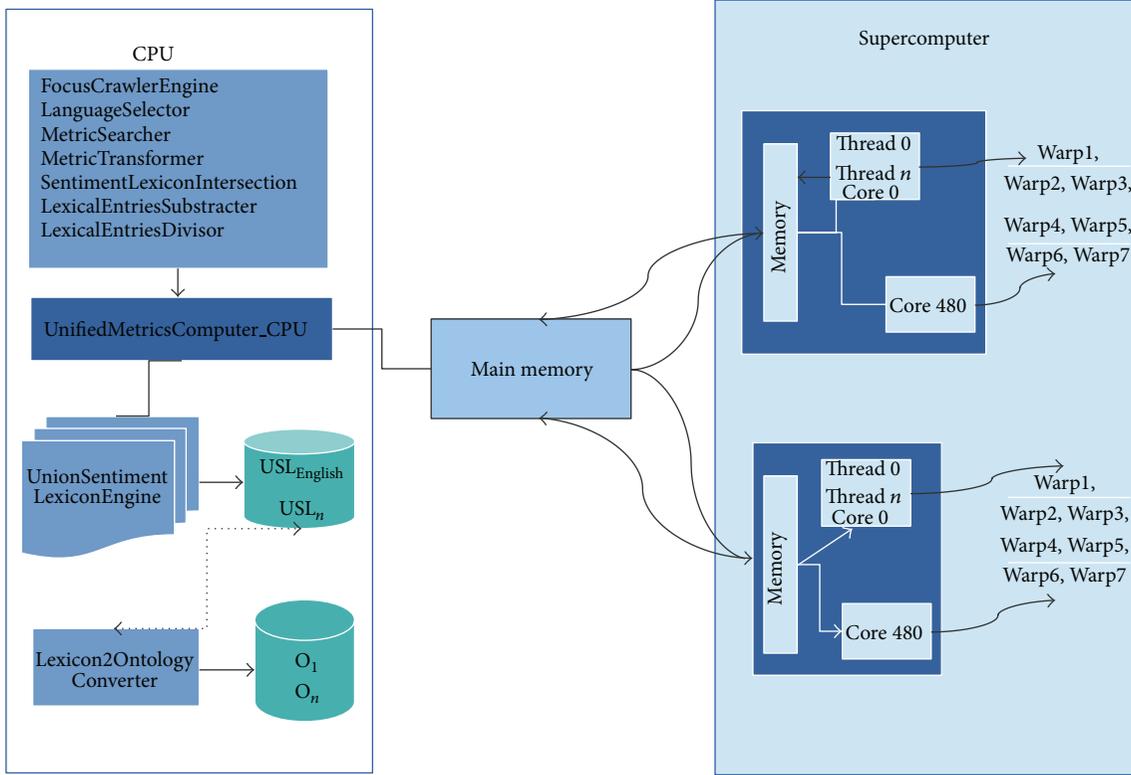


FIGURE 2: Architecture of USL approach.

as shown in Figure 1(a) and Figure 1(b); the Pan American sentiment lexicon is in Figure 1(c); SentiWordNet is in Figure 1(d); NTU Sentiment Dictionary separates into positive and negative, as shown in Figures 1(e) and 1(f); MPQA lexicon is in Figure 1(g); and Spanish Travel Subjective Lexicon is in Figure 1(h).

The second task was to filter those lexical entries that do not appear at least in two of the sentiment lexicons. The process has been described in Section 4, and an architectural view of the USL is displayed in Figure 2.

As a result, we obtained a total number of lexical entries for each sentiment lexicon: Bing Liu sentiment lexicon has 6789; MPQA lexicon contains 8221; NTU Sentiment Dictionary has 11088; Pan American has 16383; SentiWordNet has 111,711; and Spanish Travel Subjective Lexicon has 1610. The rate of each of them is displayed in Figure 3.

SentiWordNet is the lexicon with the highest number of lexical entries marked with positive polarity (12080 lexical entries); Spanish Travel Subjective Lexicon has the lowest (857 lexical entries), as shown in Figure 4(a). Bing Liu sentiment lexicon has 2006; MPQA lexicon has 2721; NTU Sentiment Dictionary has 2812; and Pan American sentiment lexicon has 6083.

In the case of neutral polarity, SentiWordNet stands out in the figure because an important subset of lexical entries (88564) has 0.0 as a strength of polarity. However, not all the sentiment lexicons assessed have neutral polarity, for example, Bing Liu sentiment lexicon, Spanish Travel Subjective Lexicon, and NTU Sentiment Dictionary; MPQA has 571

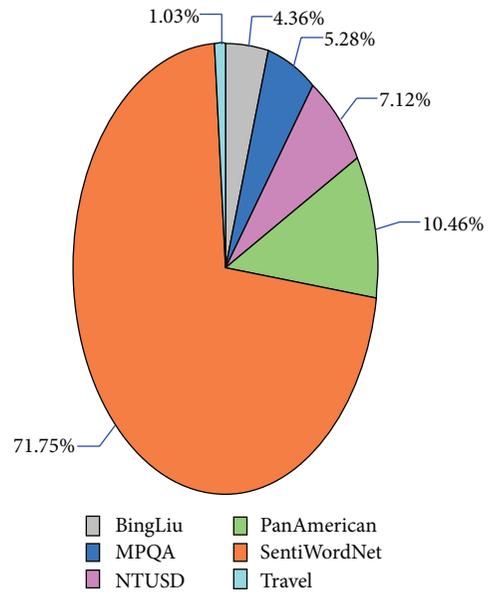


FIGURE 3: Rate of lexical entries total by sentiment lexicon.

and PanAmerican sentiment lexicon has 5000, as shown in Figure 4(b).

Figure 4(c) shows that, for negative polarity, SentiWordNet again has the highest number with 11067 lexical entries and again Spanish Travel Subjective Lexicon has the lowest

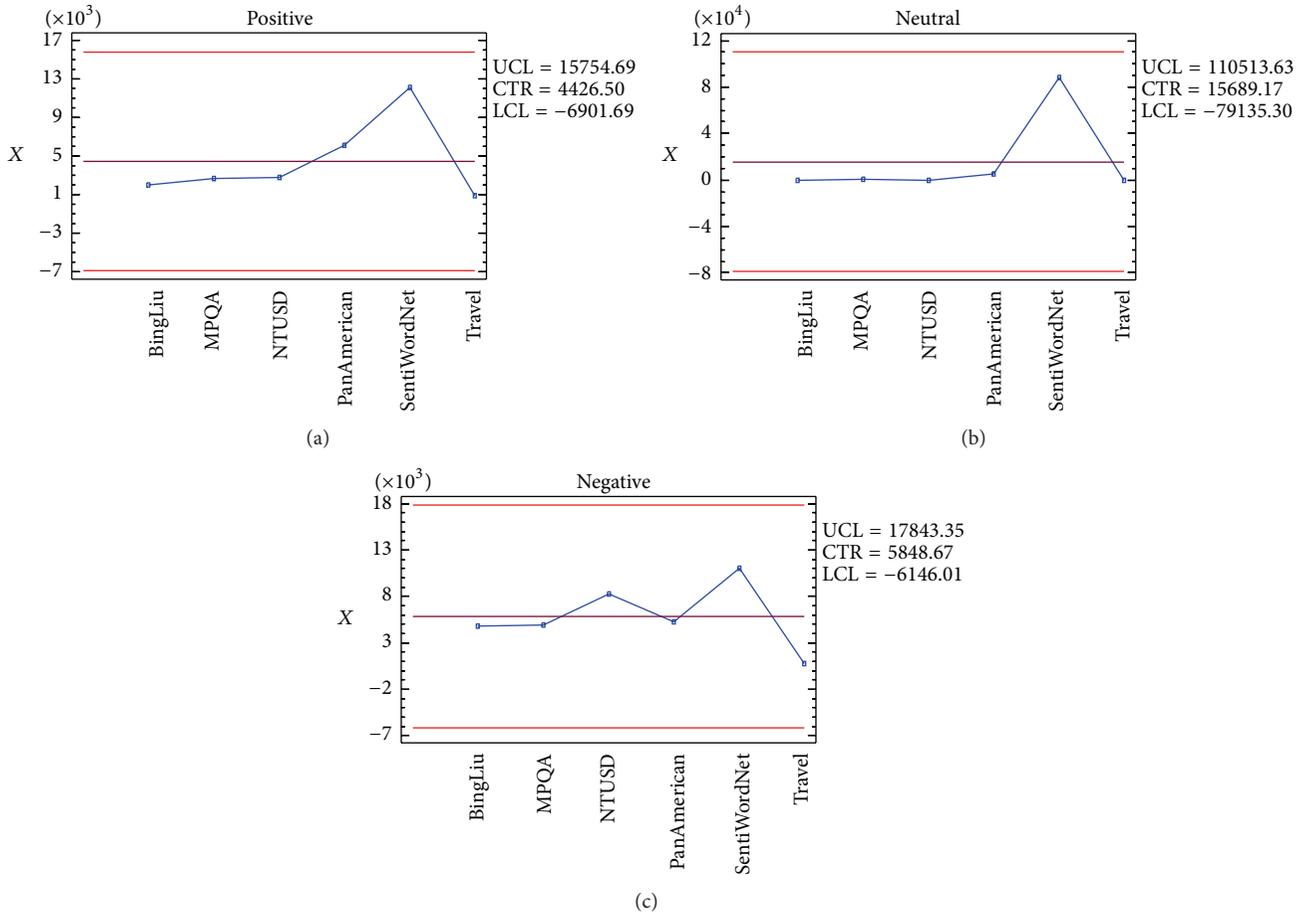


FIGURE 4: Number of lexical entries by polarity category for all the sentiment lexicons.

with 753. Bing Liu sentiment lexicon has 4783; MPQA lexicon has 4913; NTU Sentiment Dictionary has 8276; and Pan-American has 5300.

A partial result is shown in Tables 3, 4, and 5, where each table corresponds to a cluster for each language.

For the first cluster—English—a subset of lexical entries is shown in Table 3 and Figure 5.

Table 3 presents the lexical entry in each of the sentiment lexicons, the processing, and in the final column the strength of polarity of the USL.

The results are quite satisfactory although some minor problems have been detected. These problems are mainly due to the existence of expressions that can have both a positive and a negative value, and only one of the values is signalled. In the subset analysed, for instance, that is the case of the word *basic*, which can sometimes have a negative value when it is used to refer to the attributes or properties of an object or to the quality or level as in “the hotel room was too *basic*.” Another problem is the influence of the results of considering all the lexicons for the final result. That is what happens in the case of the word *achievement*. The word is correctly classified as positive, but the degree of positiveness is too low due to the fact that in SentiWordNet evaluates it with zero, thus diminishing the final score.

Table 4 shows two sentiment lexicons, STSL and Pan-American. These two lexicons have lexical entries in common, however, STSL has not assessed the strength of polarity since it has classified them simply as positive or negative. For that reason, after processing the data, the USL score obtains the strength of polarity corresponding to the only sentiment lexicon that has been assessed.

For the Chinese cluster in the first attempt of alignment with the English language a problem arose because there is not direct correspondence as different English words are represented by the same Chinese symbols.

In addition only one Chinese sentiment lexicon has been found and USL approach tries to align it with SentiWordNet but this is only for purposes of exemplification because the meanings and terms in each language are different as shown in Table 5. For example, the English word *courageous* and the Chinese word do not have exactly the same meaning. Its Chinese translation is an idiom which means “fully satisfaction.” Another case is the word *severe* which in Chinese can only be expressed by using the word meaning “*strict*.”

Another result of USL approach is OntoLexicon that was implemented in OWL language, a portion of which is shown in Algorithm 3.

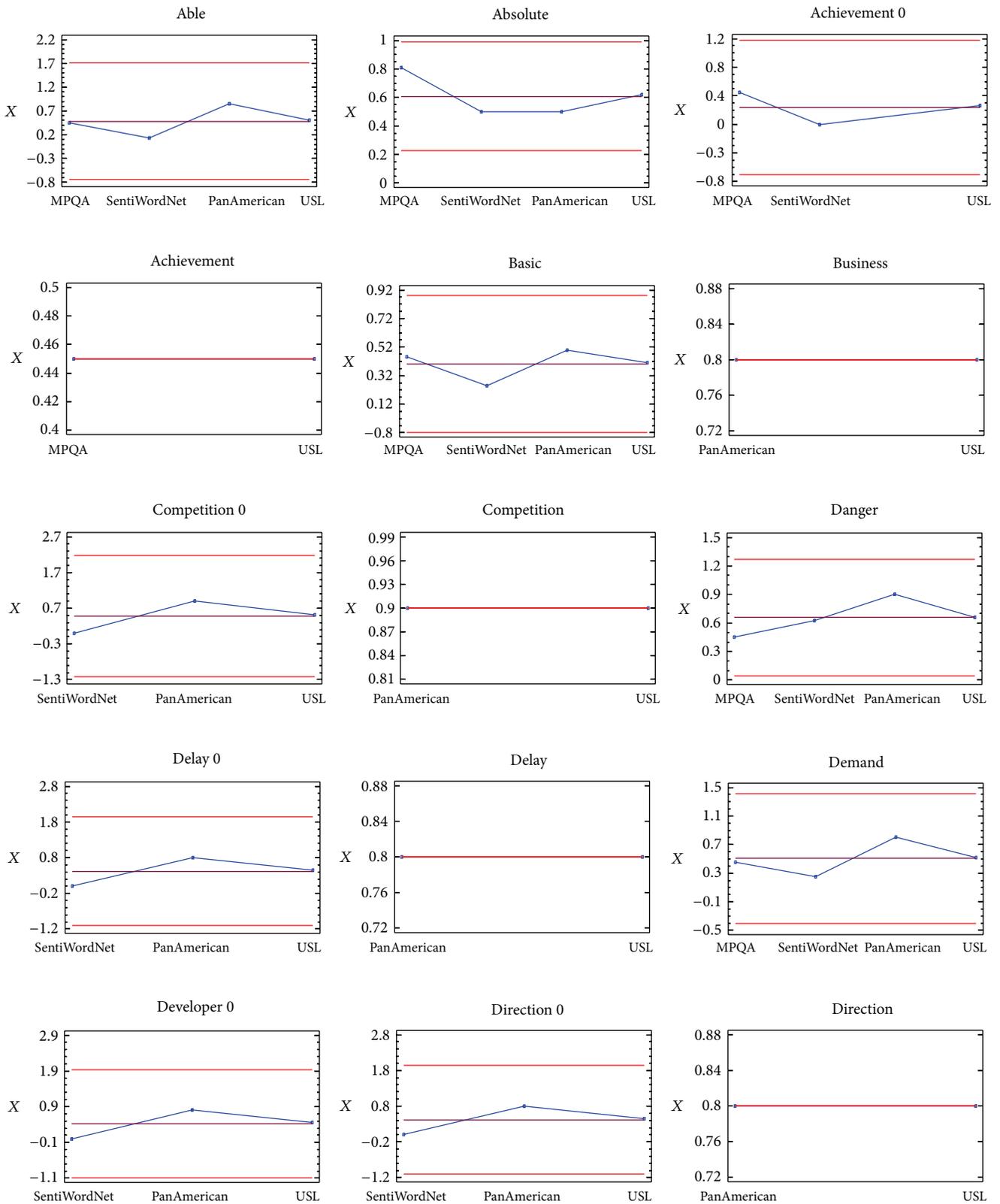


FIGURE 5: Continued.

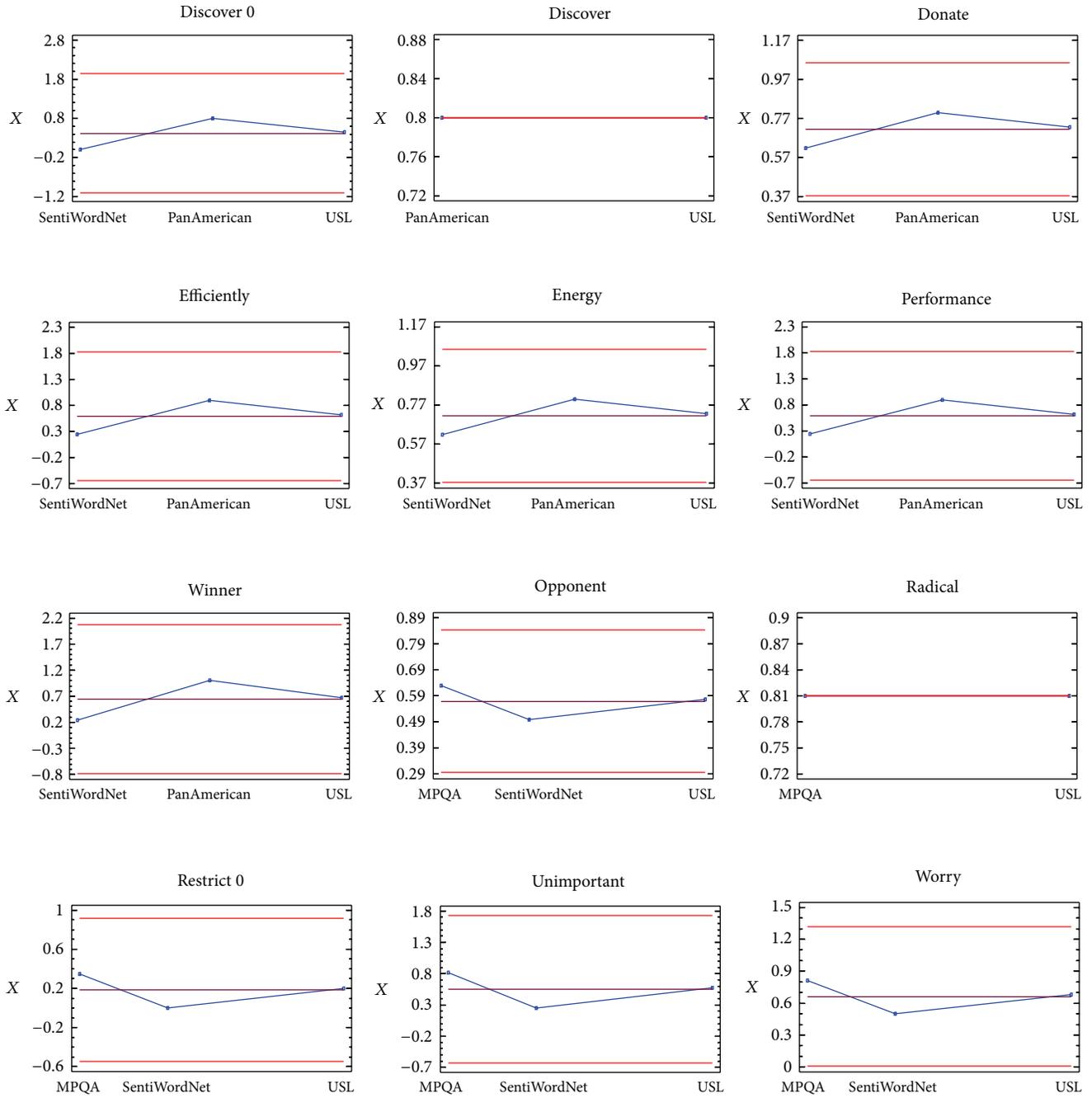


FIGURE 5: Strength of polarity for each lexical entry in Table 3 for the Unified Sentiment Lexicon, the SentyWordNet, PanAmerican, and MPQA sentiment lexicons.

The tests were performed on a CPU processor Intel XEON Hexa Core 2.50 Ghz, with 64 GiB of Ram and four GPUs: one Quadro 600 card with 96 CUDA Cores and three Tesla C2075 cards with 448 CUDA cores.

The experiment was conducted with 10, 50, 100, 500, 1,000, 5,000, and 95,430 lexical entries, respectively. As a result, the time was reduced to 3 times for the first set of data as show in Figure 6.

## 6. Discussion

It is clear that there must be progress in the task of assessing the quality of the linguistic resources. The above mentioned task takes working time (hours) on the part of expert linguists, but this work is needed if quality improvement is desired. Establishing priority criteria and stages for evaluating existing resources could help to implement this work

TABLE 3: A partial view of the USL strength of polarity for the English cluster.

Id	Lexical	Intersection	BingLiu	MPQA	SentiWordNet	PanAmerican	$\alpha$	$\beta$	$\gamma$	$\delta$	$\epsilon$	$\zeta$	USL
1		Able	X	+0.45	+0.125	+0.85	X	+0.4122	+0.0825	+0.7505	+1.2452	+2.459	<b>+0.5064</b>
2		Absolute	X	+0.81	+0.5	+0.5	X	+0.7419	+0.33	+0.4415	+1.5134	+2.459	<b>+0.6154</b>
3		Achievement	X	+0.45	0	X	X	+0.4122	0	X	+0.4122	+1.576	<b>+0.2615</b>
4		Basic	X	+0.45	+0.25	+0.5	X	+0.4122	+0.165	+0.4415	+1.0187	+2.459	<b>+0.4142</b>
5		Business	X	X	+0.8	+0.85	X	X	X	+0.7064	+0.7064	+0.883	<b>+0.8</b>
6		Competition	X	X	0	+0.9	X	X	0	+0.7947	+0.7947	+1.543	<b>+0.515</b>
7		Danger	X	-0.45	-0.625	-0.9	X	-0.4122	-0.4125	-0.7947	-1.6194	-2.459	<b>-0.6585</b>
8		Delay	X	X	0	-0.8	X	X	0	-0.7064	-0.7064	-1.543	<b>-0.4578</b>
9		Demand	X	-0.45	-0.25	-0.8	X	-0.4122	-0.165	-0.7064	-1.2836	-2.459	<b>-0.5220</b>
10		Developer	X	X	0	+0.8	X	X	0	+0.7064	+0.7064	+1.543	<b>+0.4578</b>
11		Direction	X	X	0	*0.8	X	X	0	*0.7064	*0.7064	*1.543	<b>*0.4578</b>
12		Discover	X	X	0	+0.8	X	X	0	+0.7064	+0.7064	+1.543	<b>+0.457</b>
13		Donate	X	X	+0.625	+0.8	X	X	+0.4125	+0.7064	+1.1189	+1.543	<b>+0.725</b>
14		Efficiently	X	X	+0.25	+0.9	X	X	+0.165	+0.7947	+0.9597	+1.543	<b>+0.6219</b>
15		Energy	X	X	+0.625	+0.8	X	X	+0.4125	+0.7064	+1.1189	+1.543	<b>+0.7251</b>
16		Performance	X	X	+0.25	+0.9	X	X	+0.165	+0.7947	+0.9597	+1.543	<b>+0.6219</b>
17		Winner	X	X	+0.25	+1	X	X	+0.165	+0.883	+1.048	+1.543	<b>+0.679</b>
18		Opponent	X	-0.63	-0.5	X	X	-0.5770	-0.33	X	-0.9070	-1.576	<b>-0.575</b>
19		Radical	X	-0.81	0	X	X	-0.7419	0	X	-0.7419	-1.576	<b>-0.4707</b>
20		Restrict	X	-0.35	0	X	X	-0.3206	0	X	-0.3206	-1.576	<b>-0.2034</b>
21		Unimportant	X	-0.81	-0.25	X	X	-0.7419	-0.165	X	-0.906	-1.576	<b>-0.5754</b>
22		Worry	X	-0.81	-0.5	X	X	-0.7419	-0.33	X	-1.071	-1.576	<b>-0.6801</b>
$n$		word $_n$	$v_n$	$x_n$	$y_n$	$z_n$	$\alpha_n$	$\beta_n$	$\gamma_n$	$\delta_n$	$\epsilon_n$	$\zeta_n$	<b>usl<math>_n = \epsilon_n/\zeta_n</math></b>

TABLE 4: A partial view of the USL strength of polarity for the Spanish cluster.

Id	Lexical	Intersection	STSL	PanAmerican	$\alpha$	$\beta$	$\epsilon$	$\zeta$	USL
1		Agresividad	X	+0.85	X	+0.7505	+0.7505	+0.883	<b>+0.85</b>
2		Amigo	X	+0.9	X	+0.7947	+0.7947	+0.883	<b>+0.9</b>
3		Absurdo	X	-0.9	X	-0.9	-0.9	-0.883	<b>-1.01</b>
4		Acariciar	X	+0.9	X	+0.7947	+0.7947	+0.883	<b>+0.9</b>
5		Cansado	X	-0.8	X	-0.7064	-0.7064	-0.883	<b>-0.8</b>
6		Caminata	X	+0.9	X	+0.7947	+0.7947	+0.883	<b>+0.9</b>
7		Active	X	+0.9	X	+0.7947	+0.7947	+0.883	<b>+0.9</b>
8		Celebrar	X	+0.9	X	+0.7947	+0.7947	+0.883	<b>+0.9</b>
9		Emocionar	X	+0.9	X	+0.7947	+0.7947	+0.883	<b>+0.9</b>
10		Descanso	X	+0.8	X	+0.7064	+0.7064	+0.883	<b>+0.8</b>
11		Equivocarse	X	+0.9	X	+0.7947	+0.7947	+0.883	<b>+0.9</b>
12		Admirado	X	+0.9	X	+0.7947	+0.7947	+0.883	<b>+0.9</b>
13		Aprovechar	X	+0.9	X	+0.7947	+0.7947	+0.883	<b>+0.9</b>
14		Ayudar	X	+0.9	X	+0.7947	+0.7947	+0.883	<b>+0.9</b>
15		Vencer	X	+0.9	X	+0.7947	+0.7947	+0.883	<b>+0.9</b>
16		Sudar	X	+0.9	X	+0.7947	+0.7947	+0.883	<b>+0.9</b>
17		Merecer	X	+1	X	+0.883	+0.883	+0.883	<b>+1</b>
18		Deleite	X	+0.9	X	+0.7947	+0.7947	+0.883	<b>+0.9</b>
19		Eficaz	X	+0.9	X	+0.7947	+0.7947	+0.883	<b>+0.9</b>
20		Empeorar	X	-0.9	X	-0.7947	-0.7947	-0.883	<b>-0.9</b>
21		Desilusion	X	-0.9	X	-0.7947	-0.7947	-0.883	<b>-0.9</b>
22		Desgracia	X	-0.9	X	-0.7947	-0.7947	-0.883	<b>-0.9</b>
23		Dudar	X	-0.9	X	-0.7947	-0.7947	-0.883	<b>-0.9</b>
24		Dolor	X	+0.9	X	+0.7947	+0.7947	+0.883	<b>+0.9</b>
25		Erroneo	X	-0.9	X	-0.7947	-0.7947	-0.883	<b>-0.9</b>
26		Economico	X	+0.9	X	+0.7947	+0.7947	+0.883	<b>+0.9</b>
27		Atacar	X	-0.9	X	-0.7947	-0.7947	-0.883	<b>-0.9</b>
$n$		word $_n$	$x_n$	$y_n$	$\alpha_n$	$\beta_n$	$\epsilon_n$	$\zeta_n$	<b>usl<math>_n = \epsilon_n/\zeta_n</math></b>

```

<owl:NamedIndividual rdf:ID="variedad">
  <rdf:type rdf:resource="#PositiveNouns"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:ID="corrupcion">
  <rdf:type rdf:resource="#NegativeNouns"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:ID="generoso">
  <rdf:type rdf:resource="#PositiveAdjectives"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:ID="obeso">
  <rdf:type rdf:resource="#NegativeAdjectives"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:ID="tranquilizar">
  <rdf:type rdf:resource="#PositiveVerbs"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:ID="increpar">
  <rdf:type rdf:resource="#NegativeVerbs"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:ID="a_faltar">
  <rdf:type rdf:resource="#NegativeAdverbs"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:ID="por_fortuna">
  <rdf:type rdf:resource="#PositiveAdverbs"/>
</owl:NamedIndividual>
  
```

ALGORITHM 3

TABLE 5: A partial view of the USL strength of polarity for the Chinese Cluster.

Id Lexical	Intersection	NTUSD	SentiWordNet	$\alpha$	$\beta$	$\epsilon$	$\zeta$	USL
1	(courageous)	X	+0.375	X	+0.3311	+0.3311	+0.883	<b>0.375</b>
2	(content)	X	+0.45	+0.125	+0.85	X	+0.4122	<b>+0.5064</b>
3	(agreement)	X	0	X	+0.9	+0.9	+0.883	<b>+1.019</b>
4	(perfection)	X	-0.5	X	-0.4415	-0.4415	-0.883	<b>-0.5</b>
5	(philosopher)	X	0	X	0	0	0.883	<b>0.0</b>
6	(difficulty)	X	-0.5	X	-0.4415	-0.4415	-0.883	<b>-0.5</b>
7	(sublime)	X	+0.625	X	+0.5518	+0.5518	+0.883	<b>+0.625</b>
8	(fabulous)	X	+0.875	X	+0.772	+0.772	+0.883	<b>+0.875</b>
9	(endeavour)	X	0	X	0	0	0.883	<b>0</b>
10	(good)	X	+0.625	X	+0.5518	+0.5518	+0.883	<b>+0.625</b>
11	(welcome)	X	+0.5	X	+0.4415	+0.4415	+0.883	<b>+0.5</b>
12	(praise)	X	0	X	0	0	0.883	<b>0</b>
13	(severe)	X	-0.625	X	-0.5528	-0.5518	0.883	<b>-0.625</b>
$n$	word <sub><math>n</math></sub>	$x_n$	$y_n$	$\alpha_n$	$\beta_n$	$\epsilon_n$	$\zeta_n$	<b>usl<sub><math>n</math></sub> = <math>\epsilon_n/\zeta_n</math></b>

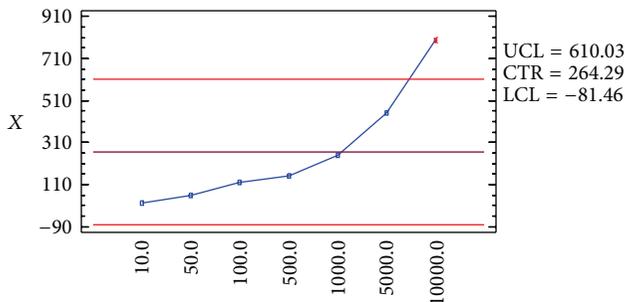


FIGURE 6: Number of gain at different scales of lexical entries.

realistically. If multilinguality is an aim then lexical resources in more languages need to be developed.

## 7. Conclusion and Future Work

We show that it is possible to unify the sentiment lexicons available on the web and align and expand them automatically. Our USL approach reuses the research work carried out by universities such as Illinois, Pittsburg, The Institute of Information Science, Taiwan, Istituto di Scienza e Tecnologie dell'Informazione, and Universidad Politécnic de Madrid.

These sentiment lexicons have been essential for the implementation of the Unified Sentiment Lexicon. Our aim is to establish the USL as a standard that could be enriched and used by the whole community in the future.

The results of the USL approach are (a) the Unified Sentiment Lexicon and (b) OntoLexicon. Using parallel processing for the calculation of strength of polarity for each lexical entry, the USL approach accelerated the runtime by 300%. USL approach avoids hard disk operations and distributes the calculation of the USL metric over 1536 processors doing operations directly in GPU memory. The unification was carried out by means of providing a single strength of polarity for each lexical entry; this value must be present at least in one intersection in two or more sentiment lexicons in the same cluster. Compared with previous work, the major contributions of this paper are the following.

- (i) A knowledge base of four sentiment lexicons (Bing Liu sentiment lexicon; MPQA; NTU Sentiment Dictionary; and SentiWordNet) has been unified automatically, grouped into three clusters—English, Spanish, and Chinese. In the final version Portuguese was not included because there are not enough sentiment lexicons available.
- (ii) The USL approach computes a unified strength of polarity which was validated by experts.
- (iii) The USL were expanded, with two additional sentiment lexicons that were developed by our research group, Communication in Specialized Domains: Pan-American sentiment lexicon and Spanish Travel Subjective Lexicon.
- (iv) The task of strength of polarity unification uses parallel processing to compute each lexical entry with GPUs.
- (v) USL computing time was accelerated 300% in data processing.
- (vi) The robustness of the Unified Sentiment Lexicon was proven with 35201 positive, 38200 neutral, and 22029 negative lexical entries.
- (vii) A uniform knowledge representation of the sentiment lexicon was made with OntoLexicon in OWL language.

Future work will involve proving the second research question: is it possible to transform a Unified Sentiment Lexicon into a generative lexicon based on a core ontology? We already have the core ontology; however, we need to transform this into a more generative lexicon. In addition, we need to extend the USL to other languages and domains, with the aim of having a unified linguistic resource to facilitate the task of subjective annotation both on the web and out of it.

## Conflict of Interests

The authors declare no conflict of interests.

## Acknowledgments

The authors are grateful to the Sciences Research Council (CONACYT) for funding this research project and they also thank NVIDIA for donating a TeslaK20 card.

## References

- [1] <http://www.census.gov/hhes/socdemo/language/data/acs/appendix.html>.
- [2] M. Lewis, G. F. S. Paul, and C. D. Fennig, *Ethnologue: Languages of the World*, SIL International, Dallas, Texas, USA, 2013.
- [3] M. R. Villarreal, "Corpus de blogs de viajes: análisis lingüístico para el reconocimiento de la valoración de la información," in *Proyecto Fin De Carrera. E. U. I. T. De Telecomunicación*, Universidad Politécnica de Madrid, 2009, (Spanish).
- [4] L. Velikovich, S. Blair-Goldensohn, K. Hannan, and R. McDonald, "The viability of web-derived polarity lexicons," in *Proceedings of the Human Language Technologies Conference of the North American Chapter of the Association for Computational Linguistics (NAACL HLT '10)*, pp. 777–785, Stroudsburg, Pa, USA, June 2010.
- [5] A. Trujillo, "Automating the lexicon: research and practice in a multilingual environment," *Natural Language Engineering*, vol. 2, no. 3, pp. 277–285, 1996.
- [6] E. Vegas, "Breadth and depth of semantic lexicons," *Computational Linguistics*, vol. 26, no. 4, pp. 652–656, 2000.
- [7] S. Staab and R. Studer, *Handbook on Ontologies*, Springer, Berlin, Germany, 2nd edition, 2009.
- [8] W. Abramowicz, Ed., *Business Information Systems*, vol. 87 of *Lecture Notes in Business Information Processing*, Springer, Berlin, Germany, 2011.
- [9] A. Pak, *Automatic, adaptive, and applicative sentiment analysis [Ph.D. thesis]*, Université Paris-Sud, Orsay, France, 2012.
- [10] M. Taboada, J. Brooke, M. Tofiloski, K. Voll, and M. Stede, "Lexicon-based methods for sentiment analysis," *Computational Linguistics*, vol. 37, no. 2, pp. 267–307, 2011.
- [11] B. Heerschoop, A. Hogenboom, and F. Frasinca, "Sentiment lexicon creation from lexical resources," *BIS*, pp. 185–196, 2011.
- [12] J. Pustejovsky, C. Havasi, J. Littman, A. Rumshisky, and M. Verhagen, "Towards a generative lexical resource: the brandeis semantic ontology," in *Proceedings of the 5th Language Resource and Evaluation Conference*, 2006.
- [13] J. Pustejovsky, "The generative lexicon," *Computational Linguistics*, vol. 17, no. 4, pp. 409–441, 1991.
- [14] S. Bergler, "Metonymy and metaphor—boundary cases and the role of a generative lexicon," in *Proceedings of the First International Workshop on Generative Approaches to the Lexicon*, Geneva, Switzerland, 2001.
- [15] R. Thomas Gruber, "toward principles for the design of ontologies used for knowledge sharing," in *Formal Ontology in Conceptual Analysis and Knowledge Representation*, Substantial Revision of Paper Presented at The International Workshop on Formal Ontology, Kluwer Academic Publishers, 1993.
- [16] A. Sheppard, *Programming GPUs. Oreilly and Associate Series*, O'Reilly Media, 2012.
- [17] J. Yan, D. B. Bracewell, F. Ren, and S. Kuroiwa, "The creation of a chinese emotion ontology based on hownet," *Engineering Letters*, vol. 16, no. 1, pp. 166–171, 2008.
- [18] J. Wiskin, D. Borup, S. Johnson et al., "Inverse scattering and refraction corrected reflection for breast cancer imaging," in

*Medical Imaging: Ultrasonic Imaging, Tomography, and Therapy*, vol. 7629 of *Proceedings of SPIE*, February 2010.

- [19] L.-W. Ku and H.-H. Chen, "Mining opinions from the web: beyond relevance retrieval," *Journal of the American Society for Information Science and Technology*, vol. 58, no. 12, pp. 1838–1850, 2007.
- [20] S. Baccianella, A. Esuli, and F. Sebastiani, "Sentiwordnet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining," in *Proceedings of the LREC*, 2010.
- [21] G. A. Miller, "Wordnet: a lexical database for english," *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995.
- [22] M. Hu and B. Liu, "Mining and summarizing customer reviews," in *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '04)*, pp. 168–177, ACM, New York, NY, USA, August 2004.
- [23] B. Liu and Hu, "Opinion observer: analyzing and comparing opinions on the web," in *Proceedings of the 14th international conference on World Wide Web (WWW '05)*, pp. 342–351, ACM, New York, NY, USA, 2005.
- [24] J. R. Martin and P. R. White, *The Language of Evaluation. Appraisal in English*, Palgrave, 2005.
- [25] D. Kirk and W. Hwu, "Programming massively parallel processors: a hands-on approach," *Applications of GPU Computing Series*, Elsevier Science, 2010.
- [26] S. Cook, "CUDA programming: a developer's guide to parallel computing with GPUs," *Applications of GPU Computing Series*, Elsevier Science, 2012.
- [27] S. P. Midkiff, "Automatic parallelization: an overview of fundamental compiler techniques," *Synthesis Lectures on Computer Architecture*, vol. 19, pp. 1–169, 2012.
- [28] J. Nickolls, I. Buck, M. Garland, and K. Skadron, "Scalable parallel programming with CUDA," *Queue*, vol. 6, no. 2, pp. 40–53, 2008.
- [29] N. Wilt, *The Cuda Handbook: a Comprehensive Guide to Gpu Programming*, Prentice Hall, 2013.
- [30] T. Brandvik and G. Pullan, "An accelerated 3D Navier-Stokes solver for flows in turbomachines," *Journal of Turbomachinery*, vol. 133, no. 2, Article ID 021025, 2010.
- [31] B. G. Levine, J. E. Stone, and A. Kohlmeyer, "Fast analysis of molecular dynamics trajectories with graphics processing units-Radial distribution function histogramming," *Journal of Computational Physics*, vol. 230, no. 9, pp. 3556–3569, 2011.
- [32] L. I. Barbosa and I. Alvarez-de-Mon-y-Rego, *Towards a Unified Sentiment Lexicon (USL) based on Graphics Processing Units (GPUs)*, 2013.
- [33] E. Lindholm, J. Nickolls, S. Oberman, and J. Montrym, "NVIDIA Tesla: a unified graphics and computing architecture," *IEEE Micro*, vol. 28, no. 2, pp. 39–55, 2008.
- [34] J. Sanders and E. Kandrot, *CUDA by Example: an Introduction to General-Purpose GPU Programming*, Addison-Wesley Professional, 1st edition, 2010.
- [35] T. Wilson, J. Wiebe, and P. Hoffmann, "Recognizing contextual polarity in phrase-level sentiment analysis," in *Proceedings of the Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP '05)*, pp. 347–354, Stroudsburg, Pa, USA, October 2005.

## Research Article

# Wolf Pack Algorithm for Unconstrained Global Optimization

Hu-Sheng Wu<sup>1,2</sup> and Feng-Ming Zhang<sup>1</sup>

<sup>1</sup> Materiel Management and Safety Engineering Institute, Air Force Engineering University, Xi'an 710051, China

<sup>2</sup> Materiel Engineering Institute, Armed Police Force Engineering University, Xi'an 710086, China

Correspondence should be addressed to Hu-Sheng Wu; wuhusheng0421@gmail.com

Received 28 June 2013; Revised 13 January 2014; Accepted 27 January 2014; Published 9 March 2014

Academic Editor: Orwa Jaber Housheya

Copyright © 2014 H.-S. Wu and F.-M. Zhang. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The wolf pack unites and cooperates closely to hunt for the prey in the Tibetan Plateau, which shows wonderful skills and amazing strategies. Inspired by their prey hunting behaviors and distribution mode, we abstracted three intelligent behaviors, scouting, calling, and besieging, and two intelligent rules, winner-take-all generation rule of lead wolf and stronger-survive renewing rule of wolf pack. Then we proposed a new heuristic swarm intelligent method, named wolf pack algorithm (WPA). Experiments are conducted on a suit of benchmark functions with different characteristics, unimodal/multimodal, separable/nonseparable, and the impact of several distance measurements and parameters on WPA is discussed. What is more, the compared simulation experiments with other five typical intelligent algorithms, genetic algorithm, particle swarm optimization algorithm, artificial fish swarm algorithm, artificial bee colony algorithm, and firefly algorithm, show that WPA has better convergence and robustness, especially for high-dimensional functions.

## 1. Introduction

Global optimization is a hot topic with applications in many areas, such as science, economy, and engineering. Generally, unconstrained global optimization problems can be formulated as follows:

$$\min \text{ or } \max f(X), \quad X = (x_1, x_2, \dots, x_n), \quad (1)$$

where  $f: R^n \rightarrow R$  is a real-valued objective function,  $X \in R^n$ , and  $n$  is the number of parameters to be optimized.

As many real-world problems are becoming increasingly complex; global optimization, especially using traditional methods, is becoming a challenging task [1]. Because of its great search space, high-dimensional global optimization problems are more difficult [2]. Fortunately, many algorithms inspired by nature have become powerful tools for these problems [3–5]. Since, with long time of biological evolution and natural selection, there are many marvelous swarm intelligence phenomena in nature, which are wonderful and can give us endless inspiration. The remarkable swarm behavior of animals such as swarming ants, schooling fish, and flocking birds has for long captivated the attention of naturalists and scientists [6]. People have developed many

intelligent optimization methods to solve complex global problems in recent decades. In 1995, inspired by social behavior and movement dynamics of birds, Kennedy proposed the particle swarm optimization algorithm (PSO) [7]. In 1996, inspired by social division and foraging behavior of ant colonies, Dorigo proposed the ant colony optimization algorithm (ACO) [8]. In 2002, inspired by foraging behavior of fish schools, Li proposed the artificial fish swarm algorithm (AFSA) [9]. In 2005, motivated by the intelligent foraging behavior of honeybee swarms, Karaboga proposed the artificial bee colony (ABC) algorithm [10]. In 2008, based on the flashing behavior of fireflies, Doctor Yang proposed firefly algorithm (FA) [11]. Researchers even give some conceptions of swarm intelligent algorithms such as rats herds algorithm, mosquito swarms algorithm, and dolphins herds algorithm [12]. Birds, fishes, ants, and bees do not have any human complex intelligence such as logical reasoning and synthetic judgment, but under the same aim, food, they stand out powerful swarm intelligence through constantly adapting environment and mutual cooperation, which give us many new ideas for solving complex problems.

The wolf pack is marvelous. Harsh living environment and constant evolution for centuries have created their

rigorous organization system and subtle hunting behavior. Wolves tactics of Mongolia cavalry in Genghis Khan period, submarine tactics of Nazi Admiral Doenitz in World War II and U.S. military wolves attack system for electronic countermeasures all highlight great charm of their swarm intelligence. [13] proposes a wolf colony algorithm (WCA) to solve the optimization problem. But the accuracy and efficiency of WCA are not good enough and easily fall into local optima, especially for high-dimensional functions. So, in this paper, we reanalyzed collaborative predation behavior and prey distribution mode of wolves and proposed a new swarm intelligence algorithm, called wolf pack algorithm (WPA); Moreover, the efficiency and robustness of the new algorithm were tested by compared experiments.

The remainder of this paper is structured as follows. In Section 2, the predation behaviors and prey distribution of wolves are analyzed. In Section 3, WPA is described. Section 4 describes the experimental setup, followed by experimental results and analysis. Finally, conclusion and future work are presented in Section 5.

## 2. System Analyzing of Wolf Pack

Wolves are gregarious animals and have clearly social work division. There is a lead wolf; some elite wolves act as scouts and some ferocious wolves in a wolf pack. They cooperate well with each other and take their respective responsibility for the survival and thriving of wolf pack.

Firstly, the lead wolf, as a leader under the law of the jungle, is always the smartest and most ferocious one. It is responsible for commanding the wolves and constantly making decision by evaluating surrounding situation and perceiving information from other wolves. These can avoid the wolves in danger and command the wolves to smoothly capture prey as soon as possible.

Secondly, the lead wolf sends some elite wolves to hunt around and look for prey in the probable scope. Those elite wolves are scouts. They walk around and independently make decision according to the concentration of smell left by prey; and higher concentration means the prey is closer to the wolves. So they always move towards the direction of getting stronger smell.

Thirdly, once a scout wolf finds the trace of prey, it will howl and report that to lead wolf. Then the lead wolf will evaluate this situation and make a decision whether to summon the ferocious wolves to round up the prey or not. If they are summoned, the ferocious wolves will move fast towards the direction of the scout wolf.

Fourthly, after capturing the prey, the prey is not distributed equitably, but in an order from the strong to the weak. That is to say that, the stronger the wolf is, the more the food it will get is. Although this distribution rule will make some weak wolf dead for lack of food, it makes sure that the wolves that have the ability to capture prey get more food so as to keep being strong and can capture more prey successfully in the next time. The rule avoids that the whole pack starves to death and ensures its continuance and proliferating. In what follows, the author made detailed description and realization for the above intelligent behaviors and rules.

## 3. Wolf Pack Algorithm

**3.1. Some Definitions.** If the predatory space of the artificial wolves is a  $N \times D$  Euclidean space,  $N$  is the number of wolves,  $D$  is the number of variables. The position of one wolf  $i$  is a vector  $\mathbf{X}_i = (x_{i1}, x_{i2}, \dots, x_{iD})$ , and  $x_{id}$  is the  $d$ th variable value of the  $i$ th artificial wolf.  $Y = f(\mathbf{X})$  represents the concentration of prey's smell perceived by artificial wolves, which is also the objective function value.

The distance between two wolves  $p$  and  $q$  is described as  $L(p, q)$ . Several distance measurements can be selected according to specific problems. For example, hamming distance can be used in WPA for 0-1 discrete optimization, while Manhattan distance (MD) and Euclidean distance (ED) can be used in WPA for continuous numerical function optimization. In this paper, we mainly discuss the latter problem, and the selection of distance measurements will be discussed in Section 4.2.1. Moreover, because the problems of maximum value and minimal value can convert to each other, only the maximum value problem is discussed in what follows.

**3.2. The Description of Intelligent Behaviors and Rules.** The cooperation between lead wolf, scout wolves, and ferocious wolves makes nearly perfect predation, while prey distribution from the strong to the weak makes the wolf pack thrives towards the direction of the prey that it most probably can be able to capture. The whole predation behavior of wolf pack is abstracted three intelligent behaviors, scouting, calling, and besieging behavior, and two intelligent rules, winner-take-all generating rule for the lead wolf and the stronger-survive renewing rule for the wolf pack.

(1) The winner-take-all generating rule for the lead wolf: the artificial wolf with the best objective function value is lead wolf. During each iteration, compare the function value of the lead wolf with the best one of other wolves; if the value of lead wolf is not better, it will be replaced. Then the best wolf becomes lead wolf. Rather than acting the three intelligent behaviors, the lead wolf directly goes into the next iteration until it is replaced by other better wolf.

(2) Scouting behavior:  $S.num$  elite wolves except the lead wolf are considered as the scout wolves; they search the solution in predatory space.  $Y_i$  is the concentration of prey smell perceived by the scout wolf  $i$ .  $Y_{lead}$  is the concentration of prey smell perceived by the lead wolf.

If  $Y_i > Y_{lead}$ , that means the scout wolf is nearer to the prey and probably captures prey, so the scout wolf  $i$  becomes lead wolf and  $Y_{lead} = Y_i$ .

If  $Y_i < Y_{lead}$ , the scout wolf  $i$ , respectively, takes a step towards  $h$  different directions; the step length is  $step_a$ . After taking a step towards the  $p$ th direction, the state of the scout wolf  $i$  is formulated below:

$$x_{id}^p = x_{id} + \sin\left(2\pi \times \frac{p}{h}\right) \times step_a^d, \quad p = \{1, 2, \dots, h\}. \quad (2)$$

It should be noted that  $h$  is different for each wolf because of their different seeking ways. So  $h$  is randomly selected in  $[h_{min}, h_{max}]$  and it must be an integer.  $Y_{i0}$  is the concentration of prey smell perceived by the scout wolf  $i$  and  $Y_{ip}$  represents

the one after it took a step towards the  $p$ th direction. If  $\max\{Y_{i1}, Y_{i2}, \dots, Y_{ih}\} > Y_{i0}$ , the wolf  $i$  steps forward and its position  $X_i$  is updated. Then repeat the above until  $Y_i > Y_{lead}$  or the maximum number of repetitions  $T_{max}$  is reached.

(3) Calling behavior: the lead wolf will howl and summon  $M\_num$  ferocious wolves to gather around the prey. Here, the position of the lead wolf is considered as the one of the prey so that the ferocious wolves aggregate towards the position of lead wolf.  $step_b$  is the step length;  $g_d^k$  is the position of artificial lead wolf in the  $d$ th variable space at the  $k$ th iteration. The position of the ferocious wolf  $i$  in the  $k$ th iterative calculation is updated according to the following equation:

$$x_{id}^{k+1} = x_{id}^k + step_b^d \cdot \frac{(g_d^k - x_{id}^k)}{|g_d^k - x_{id}^k|}. \quad (3)$$

This formula consists of two parts; the former is the current position of wolf  $i$ , which represents the foundation for prey hunting; the latter represents the aggregate tendency of other wolves towards the lead wolf, which shows the lead wolf's leadership to the wolf pack.

If  $Y_i > Y_{lead}$ , the ferocious wolf  $i$  becomes lead wolf and  $Y_{lead} = Y_i$ ; then the wolf  $i$  takes the calling behavior; If  $Y_i < Y_{lead}$ , the ferocious wolf  $i$  keeps on aggregating towards the lead wolf with a fast speed until  $L(i, l) < L_{near}$ ; the wolf takes besieging behavior.  $L(i, l)$  is the distance between the wolf  $i$  and the lead wolf  $l$ ;  $L_{near}$  is the distance determinant coefficient as a judging condition, which determine whether wolf  $i$  changes state from aggregating towards the lead wolf to besieging behavior. The different value of  $L_{near}$  will affect algorithmic convergence rate. There will be a discussion in Section 4.2.2.

Calling behavior shows information transferring and sharing mechanism in wolf pack and blends the idea of social cognition.

(4) Besieging behavior: after large-steps running towards the lead wolf, the wolves are close to the prey, then all wolves except the lead wolf will take besieging behavior for capturing prey. Now, the position of lead wolf is considered as the position of prey. In particular,  $G_d^k$  represents the position of prey in the  $d$ th variable space at the  $k$ th iteration. The position of wolf  $i$  is updated according to the following equation:

$$x_{id}^{k+1} = x_{id}^k + \lambda \cdot step_c^d \cdot |G_d^k - x_{id}^k|. \quad (4)$$

$\lambda$  is a random number uniformly distributed at the interval  $[-1, 1]$ ;  $step_c$  is the step length of wolf  $i$  when it takes besieging behavior.  $Y_{i0}$  is the concentration of prey smell perceived by the wolf  $i$  and  $Y_{ik}$  represents the one after it took this behavior. If  $Y_{i0} < Y_{ik}$ , the position  $X_i$  is updated; otherwise it not changed.

There are  $step_a$ ,  $step_b$ , and  $step_c$  in the three intelligent behaviors, and the three-step length in  $d$ th variable space should have the following relationship:

$$step_a^d = \frac{step_b^d}{2} = 2 \cdot step_c^d = S. \quad (5)$$

$S$  is step coefficient and represents the fineness degree of artificial wolf searching for prey in resolution space.

(5) The stronger-survive renewing rule for the wolf pack: the prey is distributed from the strong to the weak, which will result in some weak wolves dead. The algorithm will generate  $R$  wolves while deleting  $R$  wolves with bad objective function values. Specifically, with the help of the lead wolf's hunting experience, in the  $d$ th variable space, position of the  $i$ th one of  $R$  wolves is defined as follows:

$$x_{id} = g_d \cdot rand, \quad i = \{1, 2, \dots, R\}. \quad (6)$$

$g_d$  is the position of artificial lead wolf in the  $d$ th variable space,  $rand$  is a random number uniformly distributed at the interval  $[-0.1, 0.1]$ .

When the value of  $R$  is larger, it is better for sustaining wolf's diversity and making the algorithm have the ability to open up new resolution space. But if  $R$  is too large, the algorithm will nearly be a random search approach. Because the number and scale of prey captured by wolves are different in natural word, which will lead to different number of weak wolf dead.  $R$  is an integer and randomly selected at the interval  $[n/(2 * \beta), n/\beta]$ .  $\beta$  is the population renewing proportional coefficient.

**3.3. Algorithm Description.** As described in the previous section, WPA has three artificial intelligent behaviors and two intelligent rules. There are scouting behavior, calling behavior, and besieging behavior and winner-take-all rule for generating lead wolf and the stronger-survive renewing rule for wolf pack.

Firstly, the scouting behavior accelerates the possibility that WPA can fully traverse the solution space; Secondly, the winner-take-all rule for generating lead wolf and the calling behavior make the wolves move towards the lead wolf whose position is the nearest to the prey and most likely capturing prey. The winner-take-all rule and calling behavior also make wolves arrive at the neighborhood of the global optimum only after a few iterations elapsed, since the step of wolves in calling behavior is the largest one. Thirdly, with a small step,  $step_c$ , besieging behavior makes WPA algorithm have the ability to open up new solution space and carefully search the global optima in good solution area. Fourthly, with the help of stronger-survive renewing rule for the wolf pack, the algorithm can get several new wolves whose positions are near the best wolf, lead wolf, which allows for more latitude of search space to anchor the global optimum while keeping population diversity in each iteration.

All the above make WPA possesses superior performance in accuracy and robustness, which will be seen in Section 4.

Having discussed all the components of WPA, the important computation steps are detailed below.

**Step 1 (initialization).** Initialize the following parameters, the initial position of artificial wolf  $i$  ( $X_i$ ), the number of the wolves ( $N$ ), the maximum number of iterations ( $k_{max}$ ), the step coefficient ( $S$ ), the distance determinant coefficient ( $L_{near}$ ), the maximum number of repetitions in scouting behavior ( $T_{max}$ ), and the population renewing proportional coefficient ( $\beta$ ).

TABLE 1: Benchmark functions in experiments.

No.	Functions	Formulation	Global extremum	$D$	C	Range
1	Rosenbrock	$f(\vec{x}) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$	$f_{\min}(\vec{x}) = 0$	2	UN	(-2.048, 2.048)
2	Colville	$f(\vec{x}) = 100(x_1^2 - x_2)^2 + (x_1 - 1)^2 + (x_3 - 1)^2 + 90(x_3^2 - x_4)^2 + 10.1(x_2 - 1)^2 + (x_4 - 1)^2 + 19.8(x_2 - 1)(x_4 - 1)$	$f_{\min}(\vec{x}) = 0$	4	UN	(-10, 10)
3	Sphere	$f(\vec{x}) = \sum_{i=1}^D x_i^2$	$f_{\min}(\vec{x}) = 0$	200	US	(-100, 100)
4	Sumsquares	$f(\vec{x}) = \sum_{i=1}^D i x_i^2$	$f_{\min}(\vec{x}) = 0$	150	US	(-10, 10)
5	Booth	$f(\vec{x}) = (x_1 + 2x_2 - 7)^2 + (2x_1 + x_2 - 5)^2$	$f_{\min}(\vec{x}) = 0$	2	MS	(-10, 10)
6	Bridge	$f(\vec{x}) = \frac{\sin \sqrt{x_1^2 + x_2^2}}{\sqrt{x_1^2 + x_2^2}} + \exp\left(\frac{\cos 2\pi x_1 + \cos 2\pi x_2}{2}\right) - 0.7129$	$f_{\max}(\vec{x}) = 3.0054$	2	MN	(-1.5, 1.5)
7	Ackley	$f(\vec{x}) = -20 \exp\left(-0.2 \sqrt{\frac{1}{D} \sum_{i=1}^D x_i^2}\right) - \exp\left(\frac{1}{D} \sum_{i=1}^D \cos 2\pi x_i\right) + 20 + e$	$f_{\min}(\vec{x}) = 0$	50	MN	(-32, 32)
8	Griewank	$f(\vec{x}) = \frac{1}{4000} \sum_{i=1}^D x_i^2 - \prod_{i=1}^D \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	$f_{\min}(\vec{x}) = 0$	100	MN	(-600, 600)

$D$ : dimension; C: characteristic; U: unimodal; M: multimodal; S: separable; N: nonseparable.

*Step 2.* The wolf with best function value is considered as lead wolf. In practical computation,  $S\_num = M\_num = n - 1$ , which means that wolves except for lead wolf act with different behavior as different status. So, here, except for lead wolf, according to formula (2), the rest of the  $n - 1$  wolves firstly act as the artificial scout wolves to take scouting behavior until  $Y_i > Y_{lead}$  or the maximum number of repetition  $T_{max}$  is reached and then go to Step 3.

*Step 3.* Except for the lead wolf, the rest of the  $n - 1$  wolves secondly act as the artificial ferocious wolves and gather towards the lead wolf according to (3);  $Y_i$  is the smell concentration of prey perceived by wolf  $i$ ; if  $Y_i \geq Y_{lead}$ , go to Step 2; otherwise the wolf  $i$  continues running until  $L(i, l) \leq L_{near}$ ; then go to Step 4.

*Step 4.* The position of artificial wolves who take besieging behavior is updated according to (4).

*Step 5.* Update the position of lead wolf under the winner-take-all generating rule and update the wolf pack under the population renewing rule according to (6).

*Step 6.* If the program reaches the precision requirement or the maximum number of iterations, the position and function value of lead wolf, the problem optimal solution, will be outputted; otherwise go to Step 2.

So the flow chart of WPA can be shown as Figure 1.

## 4. Experimental Results

The ingredients of the WPA method have been described in Section 3. In this section, the design of experiments is explained, sensitivity analysis of parameters on WPA is explored, and the empirical results are reported, which

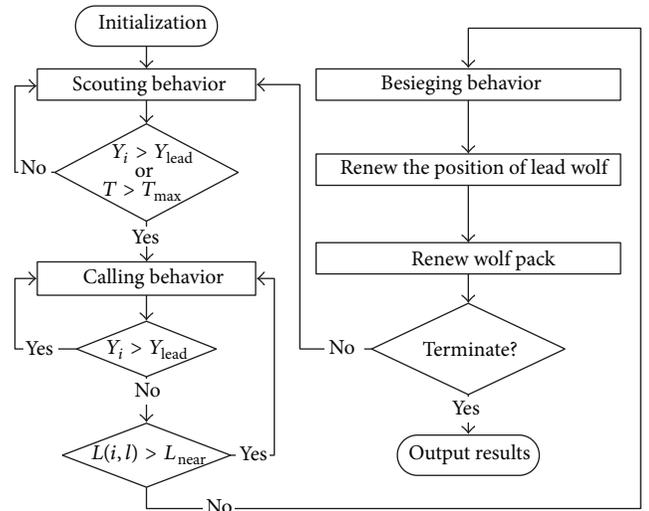


FIGURE 1: The flow chart of WPA.

compare the WPA approach with those of GA, PSO, ASFA, ABC, and FA.

### 4.1. Design of the Experiments

*4.1.1. Benchmark Functions.* In order to evaluate the performance of these algorithms, eight classical benchmark functions are presented in Table 1. Though only eight functions are used in this test, they are enough to include some different kinds of problems such as unimodal, multimodal, regular, irregular, separable, nonseparable and multidimensional.

If a function has more than one local optimum, this function is called multimodal. Multimodal functions are used to test the ability of algorithms to get rid of local minima.

TABLE 2: The list of various methods used in the paper.

Method	Authors and references
Genetic algorithm (GA)	Goldberg [14]
Particle swarm optimization algorithm (PSO)	Kennedy and Eberhart [7]
Artificial fish school algorithm (ASFA)	Li et al. [9]
Artificial bee colony algorithm (ABC)	Karaboga [10]
Firefly algorithm (FA)	Yang [11]

Another group of test problems is separable or nonseparable functions. A  $p$ -variable separable function can be expressed as the sum of  $p$  functions of one variable, such as Sumsquares and Rastrigin. Nonseparable functions cannot be written in this form, such as Bridge, Rosenbrock, Ackley, and Griewank. Because nonseparable functions have interrelation among their variable, these functions are more difficult than the separable functions.

In Table 1, characteristics of each function are given under the column titled  $C$ . In this column,  $M$  means that the function is multimodal, while  $U$  means that the function is unimodal. If the function is separable, abbreviation  $S$  is used to indicate this specification. Letter  $N$  refers to that the function is nonseparable. As seen from Table 1, 4 functions are multimodal, 4 functions are unimodal, 3 functions are separable, and 5 functions are nonseparable.

The variety of functions forms and dimensions make it possible to fairly assess the robustness of the proposed algorithms within limit iteration. Many of these functions allow a choice of dimension, and an input dimension ranging from 2 to 200 for test functions is given. Dimensions of the problems that we used can be found under the column titled  $D$ . Besides, initial ranges, formulas, and global optimum values of these functions are also given in Table 1.

**4.1.2. Experimental Settings.** In this subsection, experimental settings are given. Firstly, in order to fully compare the performance of different algorithms, we take the simulation under the same situation. So the values of the common parameters used in each algorithm such as population size and evaluation number were chosen to be the same. Population size was 100 and the maximum evaluation number was 2000 for all algorithms on all functions. Additionally, we follow the parameter settings in the original paper of GA, PSO, AFSA, ABC, and FA; see Table 2.

For each experiment, 50 independent runs were conducted with different initial random seeds. To evaluate the performance of these algorithms, six criteria are given in Table 3.

Accelerating convergence speed and avoiding the local optima have become two important and appealing goals in swarm intelligent search algorithms. So, as seen in Table 3, we adopted criteria best, mean, and standard deviation to evaluate efficiency and accuracy of algorithms and adopted criteria Art, Worst, and SR to evaluate convergence speed, effectiveness, and robustness of six algorithms.

TABLE 3: Six criteria and their abbreviations.

Criteria	Abbreviation
The best value of optima found in 50 runs	Best
The worst value of optima found in 50 runs	Worst
The average value of optima found in 50 runs	Mean
The standard deviations	StdDev
The success rate of the results	SR
The average reaching time	Art

Specifically speaking, SR provides very useful information about how stable an algorithm is. Success is claimed if an algorithm successfully gets a solution below a prespecified threshold value with the maximum number of function evaluations [15]. So, to calculate the success rate, an error accuracy level  $\varepsilon = 10^{-6}$  must be set ( $\varepsilon = 10^{-6}$  also used in [16]). Thus, we compared the result  $F$  with the known analytical optima  $F^*$  and consider  $F$  to be “successful” if the following inequality holds:

$$\frac{|F - F^*|}{F^*} < \varepsilon, \quad F^* \neq 0, \quad (7)$$

$$|F - F^*| < \varepsilon, \quad F^* = 0.$$

The SR is a percentage value that is calculated as

$$SR = \frac{\text{\#successful runs}}{\text{\#runs}}. \quad (8)$$

Art is the average value of time once an algorithm gets a solution satisfying the formula (7) in 50-run computations. Art also provides very useful information about how fast an algorithm converges to certain accuracy or under the same termination criterion, which has important practical significance.

All algorithms have been tested in Matlab 2008a over the same Lenovo A4600R computer with a Dual-Core 2.60 GHz processor, running Windows XP operating system over 1.99 Gb of memory.

**4.2. Experiments 1: Effect of Distance Measurements and Four Parameters on WPA.** In order to study the effect of two distance measures and four parameters on WPA, different measures and values of parameters were tested on typical functions listed in Table 1. Each experiment, WPA algorithm that runs 50 times on each function, and several criteria described in Section 4.1.2 are used. The experiment is conducted with the original coefficients shown in Table 9.

**4.2.1. Effect of Distance Measurements on the Performance of WPA.** This subsection will investigate the performance of different distance measurements using functions with different characteristics. As is known to all, Euclidean distance (ED) and Manhattan distance (MD) are the two most common distance metrics in practical continuous optimization. In the proposed WPA, MD or ED can be adopted to measure the distance between two wolves in the candidate solution

TABLE 4: Sensitivity analysis of distance measurements.

Function	Global extremum	$D$	Distance	Best	Worst	Mean	StdDev	SR/%	Art/s
Rosenbrock	$f_{\min}(\vec{x}) = 0$	2	MD	$9.21e - 11$	$3.24e - 8$	$1.12e - 8$	$1.18e - 8$	100	10.5165
			ED	$4.26e - 9$	$2.71e - 7$	$1.27e - 7$	$6.81e - 8$	100	37.1053
Colville	$f_{\min}(\vec{x}) = 0$	4	MD	$5.62e - 8$	$5.28e - 7$	$2.49e - 7$	$2.23e - 7$	100	46.8619
			ED	$1.74e - 7$	$1.70e - 6$	$5.74e - 7$	$3.70e - 7$	90	68.3220
Sphere	$f_{\min}(\vec{x}) = 0$	200	MD	$3.20e - 161$	$3.29e - 144$	$2.07e - 145$	$7.49e - 145$	100	11.5494
			ED	$1.76e - 160$	$3.36e - 143$	$1.68e - 144$	$7.51e - 144$	100	11.6825
Sumsquares	$f_{\min}(\vec{x}) = 0$	150	MD	$1.56e - 161$	$3.09e - 144$	$1.79e - 145$	$6.95e - 145$	100	8.5565
			ED	$3.97e - 160$	$2.24e - 144$	$1.13e - 145$	$5.00e - 145$	100	8.7109
Booth	$f_{\min}(\vec{x}) = 0$	2	MD	$5.63e - 12$	$1.15e - 10$	$4.19e - 11$	$3.32e - 11$	100	11.1074
			ED	$1.08e - 9$	$2.64e - 8$	$1.16e - 8$	$6.93e - 9$	100	40.5546
Bridge	$f_{\max}(\vec{x}) = 3.0054$	2	MD	3.0054	3.0054	3.0054	$4.56e - 16$	100	1.1093
			ED	3.0054	3.0054	3.0054	$4.56e - 16$	100	1.9541
Ackley	$f_{\min}(\vec{x}) = 0$	50	MD	$8.88e - 16$	$8.88e - 16$	$8.88e - 16$	0	100	19.3648
			ED	$8.88e - 16$	$8.88e - 16$	$8.88e - 16$	0	100	43.6884
Griewank	$f_{\min}(\vec{x}) = 0$	100	MD	0	0.1507	$3.01e - 3$	0.0213	98	$>8.77e3$
			ED	0	0.8350	0.0167	0.1181	92	$>1.35e + 4$

space. Therefore, a discussion about their impacts on the performance of WPA is needed.

There are two wolves:  $\mathbf{X}_p = (x_{p1}, x_{p2}, \dots, x_{pD})$  is the position of wolf  $p$ ,  $\mathbf{X}_q = (x_{q1}, x_{q2}, \dots, x_{qD})$  is the position of wolf  $q$ , and the ED and MD between them can be, respectively, calculated as formula (9).  $D$  is the dimension number of solution space

$$L_{ED}(p, q) = \sum_{d=1}^D (x_{pd} - x_{qd})^2, \quad (9)$$

$$L_{MD}(p, q) = \sum_{d=1}^D |x_{pd} - x_{qd}|.$$

The statistical results obtained by WPA after 50-run computation are shown in Table 4. Firstly, we note that WPA with Euclidean distance (WPA\_ED) does not get 100% success rate on Colville ( $D = 4$ ) and Griewank functions ( $D = 100$ ), while WPA with Manhattan distance (WPA\_MD) does not get 100% success rate on Griewank functions ( $D = 100$ ), which means that WPA\_ED and WPA\_MD with original coefficients still have the risk of premature convergence to local optima.

As seen from Table 4, WPA is not very sensitive to two distance measurements on most functions (Rosenbrock, Sphere, Sumsquares, Booth, and Ackley), and no matter which metric is used, WPA can always get a good result with SR = 100%. But, for these functions, comparing the results between WPA\_MD and WPA\_ED in detail, we can find that WPA\_MD has shorter average reaching time (ART), which means faster convergence speed to a certain accuracy. The reason may be that ED has the higher computational complexity. Meanwhile, WPA\_MD has better performance on other four criteria (best, worst, mean, and StdDev), which means better solution accuracy and robustness.

Naturally, because of its better efficiency, precision, and robustness, WD is more suitable for WPA. So the WPA algorithm used in what follows is WPA\_MD.

*4.2.2. Effect of Four Parameters on the Performance of WPA.* In this subsection, we investigate the impact of the parameters  $S$ ,  $L_{\text{near}}$ ,  $T_{\text{max}}$ , and  $\beta$  on the new algorithm.  $S$  is the step coefficient,  $L_{\text{near}}$  is the distance determinant coefficient,  $T_{\text{max}}$  is the maximum number of repetitions in scouting behavior, and  $\beta$  is the population renewing proportional coefficient. The parameters selection procedure is performed in a one-factor-at-a-time manner. For each sensitivity analysis in this section, only one parameter is varied each time, and the remaining parameters are kept at the values suggested by the original estimate listed in Table 9. The interaction relation between parameters is assumed unimportant.

Each time one of the WPA parameters is varied in a certain interval to see which value within this interval will result in the best performance. Specifically, the WPA algorithm also runs 50 times on each case.

Table 5 shows the sensitivity analysis of the step coefficient  $S$ . All results are shown in the form of Mean  $\pm$  Std (SR/%). The choice of interval [0.04, 0.16] used in this analysis was motivated by the original Nelder-Mead simplex search procedure, where a step coefficient greater than 0.04 was suggested for general usage.

Meanwhile, based on detailed comparison of the results, on Rosenbrock, Sphere, and Bridge functions, step coefficient is not sensitive to WPA, and for Booth function there is a tendency of better results with larger  $S$ . From Table 5, it is found that a step coefficient setting at 0.12 returns the best result which has better Mean, small Std, and SR = 100% for all functions.

Tables 6–8 analyze sensitivity of  $L_{\text{near}}$ ,  $T_{\text{max}}$ , and  $\beta$ . Generally speaking,  $L_{\text{near}}$ ,  $T_{\text{max}}$ , and  $\beta$  are not sensitive to most functions except Griewank function, since Griewank not only

TABLE 5: Sensitivity analysis of step coefficient (S).

Functions	Mean $\pm$ Std (SR/%) (the default of SR is 100%)						
	0.04	0.06	0.08	0.10	0.12	0.16	
Rosenbrock	6.9e - 8 $\pm$ 4.3e - 8	2.7e - 8 $\pm$ 3.5e - 8	1.1e - 8 $\pm$ 9.1e - 9	3.2e - 9 $\pm$ 2.7e - 9	5.0e - 9 $\pm$ 5.7e - 9	3.2e - 9 $\pm$ 3.7e - 9	1.2e - 9 $\pm$ 1.6e - 9
Colville	1.3e - 7 $\pm$ 7.1e - 8	3.3e - 7 $\pm$ 2.8e - 7 (90)	2.6e - 7 $\pm$ 1.9e - 7	2.3e - 7 $\pm$ 1.4e - 7	3.5e - 7 $\pm$ 2.5e - 7	9.5e - 7 $\pm$ 1.0e - 6 (80)	1.4e - 6 $\pm$ 1.5e - 6 (50)
Sphere	2.3e - 145 $\pm$ 7.1e - 145	6.6e - 152 $\pm$ 2.1e - 151	2.1e - 146 $\pm$ 4.5e - 146	3.9e - 146 $\pm$ 1.2e - 145	1.2e - 145 $\pm$ 3.4e - 145	1.7e - 146 $\pm$ 5.3e - 146	2.2e - 149 $\pm$ 6.8e - 149
Sumsquares	9.8e - 145 $\pm$ 3.1e - 144	3.1e - 146 $\pm$ 8.4e - 146	8.1e - 147 $\pm$ 2.6e - 146	4.8e - 146 $\pm$ 1.0e - 145	3.8e - 152 $\pm$ 7.9e - 152	3.4e - 147 $\pm$ 1.1e - 146	1.2e - 147 $\pm$ 3.9e - 147
Booth	5.4e - 7 $\pm$ 3.3e - 7	1.6e - 9 $\pm$ 1.1e - 9	3.2e - 11 $\pm$ 1.6e - 11	1.3e - 12 $\pm$ 9.1e - 13	1.3e - 13 $\pm$ 1.2e - 13	3.9e - 15 $\pm$ 1.8e - 15	1.2e - 16 $\pm$ 5.8e - 17
Bridge	3.0054 $\pm$ 4.7e - 16	3.0054 $\pm$ 4.7e - 16	3.0054 $\pm$ 4.7e - 16	3.0054 $\pm$ 4.7e - 16	3.0054 $\pm$ 4.7e - 16	3.0054 $\pm$ 4.7e - 16	3.0054 $\pm$ 4.7e - 16
Ackley	8.9e - 16 $\pm$ 0	0.25 $\pm$ 0.53 (80)	1.2e - 15 $\pm$ 1.1e - 15	8.9e - 16 $\pm$ 0	1.2e - 15 $\pm$ 1.1e - 15	8.9e - 16 $\pm$ 0	8.9e - 16 $\pm$ 0
Griewank	0 $\pm$ 0	0 $\pm$ 0	0 $\pm$ 0	0 $\pm$ 0	0 $\pm$ 0	0.06 $\pm$ 0.19 (92)	0.20 $\pm$ 0.42 (86)

TABLE 6: Sensitivity analysis of distance determinant coefficient ( $L_{\text{near}}$ ).

Functions	Mean $\pm$ Std (SR/%) (the default of SR is 100%)									
	0.04	0.06	0.08	0.10	0.12	0.14	0.16			
Rosenbrock	$4.4e-8 \pm 6.5e-8$	$2.3e-8 \pm 3.7e-8$	$3.4e-9 \pm 4.8e-9$	$3.0e-8 \pm 2.9e-8$	$1.9e-8 \pm 2.4e-8$	$2.4e-8 \pm 4.7e-8$	$2.9e-8 \pm 5.3e-8$			
Colville	$2.0e-7 \pm 9.9e-8$	$2.6e-7 \pm 1.6e-7$	$3.5e-7 \pm 2.6e-7$	$2.3e-7 \pm 1.5e-7$	$1.2e-7 \pm 3.4e-8$	$2.8e-7 \pm 1.9e-7$	$1.4e-7 \pm 6.9e-8$			
Sphere	$6.8e-146 \pm 2.0e-145$	$1.9e-146 \pm 6.2e-146$	$1.7e-145 \pm 4.3e-145$	$2.6e-148 \pm 8.3e-148$	$3.6e-146 \pm 1.1e-145$	$3.7e-151 \pm 1.1e-150$	$5.3e-149 \pm 1.7e-148$			
Sumsquares	$1.1e-147 \pm 3.4e-147$	$1.0e-146 \pm 3.3e-146$	$3.7e-151 \pm 8.9e-151$	$6.2e-146 \pm 1.9e-145$	$6.2e-152 \pm 1.9e-151$	$1.22e-145 \pm 2.9e-145$	$1.3e-148 \pm 4.0e-148$			
Booth	$2.6e-11 \pm 1.3e-11$	$2.9e-11 \pm 1.9e-11$	$2.4e-11 \pm 1.6e-11$	$3.1e-11 \pm 1.8e-11$	$2.4e-11 \pm 1.3e-11$	$3.1e-11 \pm 2.1e-11$	$1.0e-10 \pm 1.3e-10$			
Bridge	$3.0054 \pm 4.7e-16$	$3.0054 \pm 4.7e-16$	$3.0054 \pm 4.7e-16$	$3.0054 \pm 4.7e-16$	$3.0054 \pm 4.7e-16$	$3.0054 \pm 4.7e-16$	$3.0054 \pm 4.7e-16$			
Ackley	$0.14 \pm 0.43$ (90)	$1.2e-15 \pm 1.1e-15$	$8.9e-16 \pm 0$	$1.2e-15 \pm 1.1e-15$	$8.9e-16 \pm 0$	$1.59e-15 \pm 1.49e-15$	$8.9e-16 \pm 0$			
Griewank	$0.08 \pm 0.26$ (90)	$1.0e-3 \pm 0.02$ (96)	$0 \pm 0$	$0 \pm 0$	$0 \pm 0$	$0.10 \pm 0.33$ (92)	$0 \pm 0$			

TABLE 7: Sensitivity analysis of the maximum number of repetitions in scouting behavior ( $T_{\max}$ ).

Functions	Mean $\pm$ Std (SR/%) (the default of SR is 100%)								
	6	8	10	12	14	16	18		
Rosenbrock	2.4e - 8 $\pm$ 2.6e - 8	8.4e - 9 $\pm$ 8.0e - 9	1.3e - 8 $\pm$ 1.3e - 8	1.4e - 8 $\pm$ 1.0e - 8	2.0e - 8 $\pm$ 1.9e - 8	2.1e - 8 $\pm$ 2.5e - 8	1.2e - 8 $\pm$ 8.9e - 9		
Colville	4.8e - 7 $\pm$ 2.2e - 7	3.4e - 7 $\pm$ 1.8e - 7	1.5e - 7 $\pm$ 1.2e - 7	3.8e - 7 $\pm$ 2.0e - 7	3.6e - 7 $\pm$ 3.7e - 7 (96)	3.4e - 7 $\pm$ 2.5e - 7	2.6e - 7 $\pm$ 1.5e - 7		
Sphere	7.1e - 147 $\pm$ 2.2e - 146	4.5e - 146 $\pm$ 9.0e - 146	7.8e - 146 $\pm$ 2.3e - 145	1.9e - 148 $\pm$ 5.3e - 148	5.7e - 148 $\pm$ 1.3e - 147	6.9e - 145 $\pm$ 2.2e - 144	3.6e - 147 $\pm$ 1.1e - 146		
Sumsquares	4.1e - 146 $\pm$ 1.3e - 145	2.4e - 149 $\pm$ 4.8e - 149	4.2e - 149 $\pm$ 1.3e - 148	8.3e - 150 $\pm$ 2.6e - 149	8.5e - 147 $\pm$ 2.7e - 146	5.4e - 146 $\pm$ 9.0e - 146	1.4e - 151 $\pm$ 4.4e - 151		
Booth	3.2e - 11 $\pm$ 2.9e - 11	4.2e - 11 $\pm$ 2.7e - 11	2.5e - 11 $\pm$ 1.5e - 11	2.1e - 11 $\pm$ 1.5e - 11	3.2e - 11 $\pm$ 2.5e - 11	2.6e - 11 $\pm$ 1.8e - 11	2.6e - 11 $\pm$ 2.7e - 11		
Bridge	3.0054 $\pm$ 4.7e - 16	3.0054 $\pm$ 4.7e - 16	3.0054 $\pm$ 4.7e - 16	3.0054 $\pm$ 4.7e - 16	3.0054 $\pm$ 4.7e - 16	3.0054 $\pm$ 4.7e - 16	3.0054 $\pm$ 4.7e - 16		
Ackley	8.9e - 16 $\pm$ 0	8.9e - 16 $\pm$ 0	8.9e - 16 $\pm$ 0	8.9e - 16 $\pm$ 0	1.2e - 15 $\pm$ 1.1e - 15	8.9e - 16 $\pm$ 0	8.9e - 16 $\pm$ 0		
Griewank	0.10 $\pm$ 0.33 (92)	0 $\pm$ 0	1.0e - 3 $\pm$ 0.02 (98)	0.09 $\pm$ 0.31 (88)	0 $\pm$ 0	0.09 $\pm$ 0.29 (94)	8.3e - 4 $\pm$ 0.02 (98)		

TABLE 8: Sensitivity analysis of population renewing proportional coefficient ( $\beta$ ).

Functions	Mean $\pm$ Std (SR/%) (the default of SR is 100%)							
	2	3	4	5	6	7	8	
Rosenbrock	$1.0e - 8 \pm 9.2e - 9$	$8.7e - 9 \pm 7.6e - 9$	$1.2e - 8 \pm 1.0e - 8$	$8.6e - 9 \pm 8.3e - 9$	$1.4e - 8 \pm 1.3e - 8$	$9.9e - 9 \pm 9.8e - 9$	$1.1e - 8 \pm 1.2e - 9$	
Colville	$3.2e - 8 \pm 1.8e - 8$	$1.4e - 7 \pm 1.3e - 7$	$1.2e - 7 \pm 5.9e - 8$	$1.4e - 7 \pm 9.4e - 8$	$3.0e - 7 \pm 6.9e - 8$	$3.9e - 7 \pm 1.7e - 7$	$8.6e - 7 \pm 4.0e - 7$ (80)	
Sphere	$1.9e - 166 \pm 0$	$5.2e - 158 \pm 1.6e - 157$	$2.9e - 153 \pm 9.2e - 153$	$4.3e - 149 \pm 1.3e - 148$	$7.9e - 139 \pm 2.5e - 138$	$8.3e - 134 \pm 1.8e - 133$	$3.4e - 126 \pm 8.0e - 126$	
Sumsquares	$2.8e - 167 \pm 0$	$1.4e - 157 \pm 4.3e - 157$	$2.8e - 155 \pm 4.5e - 155$	$8.3e - 146 \pm 1.8e - 145$	$6.9e - 143 \pm 1.7e - 142$	$5.3e - 143 \pm 1.3e - 142$	$3.3e - 127 \pm 1.0e - 126$	
Booth	$8.1e - 11 \pm 1.3e - 10$	$2.5e - 11 \pm 1.7e - 11$	$1.9e - 11 \pm 1.2e - 11$	$2.5e - 11 \pm 1.7e - 11$	$2.5e - 11 \pm 1.5e - 11$	$2.3e - 11 \pm 1.5e - 11$	$2.3e - 11 \pm 1.4e - 11$	
Bridge	$3.0054 \pm 4.7e - 16$	$3.0054 \pm 4.7e - 16$	$3.0054 \pm 4.7e - 16$	$3.0054 \pm 4.7e - 16$	$3.0054 \pm 4.7e - 16$	$3.0054 \pm 4.7e - 16$	$3.0054 \pm 4.7e - 16$	
Ackley	$8.9e - 16 \pm 0$	$8.9e - 16 \pm 0$	$8.9e - 16 \pm 0$	$8.9e - 16 \pm 0$	$8.9e - 16 \pm 0$	$8.9e - 16 \pm 0$	$8.9e - 16 \pm 0$	
Griewank	$0 \pm 0$	$0 \pm 0$	$0 \pm 0$	$0 \pm 0$	$0.19 \pm 0.41$ (86)	$0 \pm 0$	$1.2e - 3 \pm 0.31$ (96)	

TABLE 9: Best suggestions for WPA parameters.

No.	WPA parameters name	Original	Best-suggested
1	Step coefficient ( $S$ )	0.08	0.12
2	Distance determinant coefficient ( $L_{near}$ )	0.12	0.08
3	The maximum number of repetitions in scouting ( $T_{max}$ )	10	8
4	Population renewal coefficient ( $\beta$ )	5	2

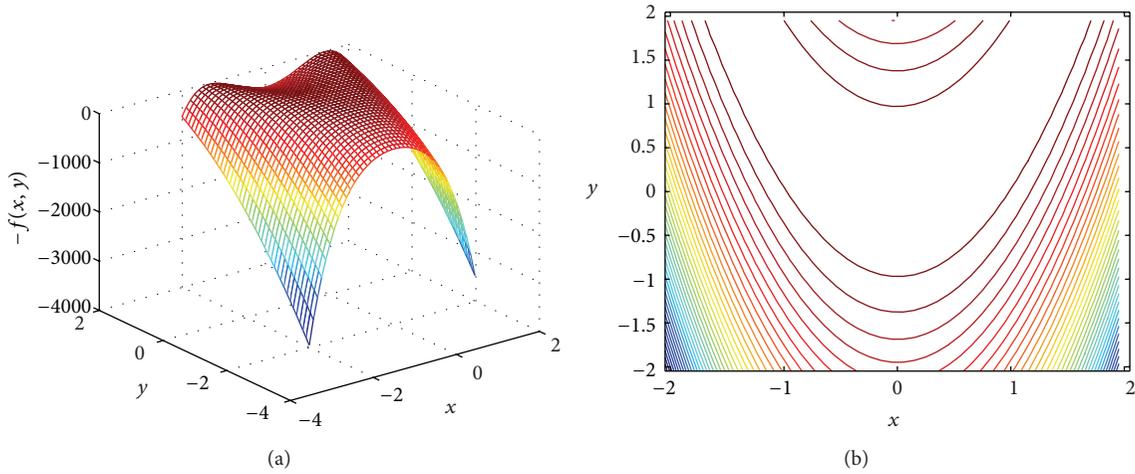


FIGURE 2: Rosenbrock function ( $D = 2$ ): (a) surface plot and (b) contour lines.

is a high-dimensional function for its 100 parameters, but also has very large search space for its interval of  $[-600, 600]$ , which is hard to optimized.

Table 6 illustrates the sensitivity analysis of  $L_{near}$ , and from this table it is found that setting  $L_{near}$  at 0.08 returns the best results with the best mean, smaller standard deviations, and 100% success rate for all functions.

Tables 7-8 indicate that  $T_{max}$  and  $\beta$ , respectively, setting at 8 and 2 return the best results on eight functions.

So we summarize the above findings in Table 9 and apply these parameter values in our approach for conducting experimental comparisons with other algorithms listed in Table 2.

**4.3. Experiments 2: WPA versus GA, PSO, AFSA, ABC, and FA.** In this section, we compared GA, PSO, AFSA, ABC, FA, and WPA algorithms on eight functions described in Table 1. Each of the experiments was repeated for 50 runs with different random seeds and the best, worst, and mean values, standard deviations, success rates, and average reaching time are given in Table 10. The best results for each case are highlighted in boldface.

As can clearly be seen from Table 10, when solving the unimodal nonseparable problems (Rosenbrock, Colville), although the results of WPA are not good enough as FA or ASFA algorithm, WPA also achieves 100% success rate. Firstly, with respect to Rosenbrock function, its surface plot and contour lines are shown in Figure 2.

As seen in Figure 2, Rosenbrock function is well known for its Rosenbrock valley. Global minimum value for this function is 0 and optimum solution is  $(x_1, x_2) = (1, 1)$ .

But the global optimum is inside a long, narrow, parabolic-shaped flat valley. Since it is difficult to converge to the global optimum of this function, the variables are strongly dependent, and the gradients generally do not point towards the optimum; this problem is repeatedly used to test the performance of the algorithms [17]. As shown in Table 10, PSO, AFSA, FA, and WPA achieve 100% success rate, and PSO shows the fastest convergence speed; AFSA gets the value  $1.10e - 13$  with the best accuracy. FA also shows good performance because of its robustness on Rosenbrock function.

On the Colville function, its surface plot and contour lines are shown in Figure 3. Colville function also has a narrow curving valley and it is hard to be optimized if the search space cannot be explored properly and the direction changes cannot be kept up with. Its global minimum value is 0 and optimum solution is  $(x_1, x_2, x_3, x_4) = (1, 1, 1, 1)$ .

Although the best accurate solution is obtained by AFSA, WPA outperforms the other algorithms in terms of the worst, mean, std., SR, and Art on Colville function.

Sphere and Sumsquares are convex, unimodal, and separable functions. They are all high-dimensional functions for their 200 and 150 parameters, respectively, and the global minima are all 0 and optimum solution is  $(x_1, x_2, \dots, x_m) = (0, 0, \dots, 0)$ . Surface plot and contour lines of them are, respectively, shown in Figures 4 and 5.

As seen from Table 10, when solving the unimodal separable problems, we note that WPA outperforms other five algorithms both on convergence speed and solution accuracy. In particular, WPA offers the highest accuracy and improves the precision by about 170 orders of magnitude on Sphere and

TABLE 10: Statistical results of 50 runs obtained by GA, PSO, AFSA, ABC, FA, and WPA algorithms.

Function	Global extremum	D	C	Algorithms	Best	Worst	Mean	StdDev	SR/%	Art/s
Rosenbrock	$f_{\min}(\vec{x}) = 0$	2	UN	GA	1.78e-10	0.0373	0.0091	0.0092	10	>759,8323
				PSO	2.26e-11	5.89e-7	1.07e-7	1.30e-7	100	0.7444
				AFSA	1.10e-13	1.11e-9	2.34e-10	2.62e-10	100	2.0578
				ABC	5.99e-6	0.0099	8.61e-4	0.0015	0	>391.0297
				FA	6.28e-13	6.29e-10	1.86e-10	1.62e-10	100	33.1256
				WPA	3.49e-11	2.34e-8	5.09e-9	4.34e-9	100	6.6333
Colville	$f_{\min}(\vec{x}) = 0$	4	UN	GA	0.0022	0.3343	0.1272	0.1062	0	>1.22e+3
				PSO	1.29e-6	3.46e-4	5.06e-5	6.71e-5	0	>114.0869
				AFSA	3.66e-8	8.91e-7	3.16e-7	2.32e-7	100	40.1807
				ABC	0.0103	0.5337	0.1871	0.1232	0	>384.4193
				FA	2.41e-7	3.69e-5	6.62e-6	8.07e-6	8	>3.14e+3
				WPA	4.71e-8	3.72e-7	1.25e-7	6.97e-8	100	27.4054
Sphere	$f_{\min}(\vec{x}) = 0$	200	US	GA	1.56e+5	1.81e+5	1.71e+5	5.78e+3	0	>4.44e+4
				PSO	1.0361	1.5520	1.2883	0.1206	0	>271.9201
				AFSA	5.12e+5	5.79e+5	5.51e+5	1.63e+4	0	>7.41e+3
				ABC	0.0041	1.2521	0.0444	0.1773	0	>442.9045
				FA	0.1432	0.2327	0.1865	0.0199	0	>8.34e+3
				WPA	1.49e-172	2.41e-165	1.56e-166	0	100	6.1729
Sumsquares	$f_{\min}(\vec{x}) = 0$	150	US	GA	5.93e+4	7.15+4	6.63e+4	2.88e+3	0	>3.16e+4
				PSO	39.7098	91.1145	55.9050	10.4165	0	>232.5464
				AFSA	1.43e+5	1.79e+5	1.64e+5	9.58e+3	0	>7.36e+3
				ABC	1.71e-5	0.0017	1.99e-4	3.36e-4	0	>435.1848
				FA	8.9920	99.8861	40.5721	19.2743	0	>6.88e+3
				WPA	2.68e-172	5.47e-166	2.62e-167	0	100	6.5954
Booth	$f_{\min}(\vec{x}) = 0$	2	MS	GA	4.55e-11	4.55e-11	4.55e-11	0	100	1.2621
				PSO	1.22e-12	2.41e-8	2.80e-9	4.52e-9	100	0.2079
				AFSA	3.02e-12	1.45e-9	4.61e-10	4.08e-10	100	4.4329
				ABC	6.05e-20	1.41e-17	4.63e-18	4.14e-18	100	0.4175
				FA	1.80e-12	4.39e-9	1.18e-9	1.11e-9	100	37.9191
				WPA	8.22e-15	7.05e-13	1.21e-13	1.19e-13	100	6.9339
Bridge	$f_{\max}(\vec{x}) = 3.0054$	2	MIN	GA	3.0054	3.0054	3.0054	1.35e-15	100	0.1927
				PSO	3.0054	3.0054	3.0054	4.84e-8	100	0.0929
				AFSA	3.0054	3.0047	3.0052	1.69e-4	12	>8.01e+3
				ABC	3.0054	3.0054	3.0054	3.59e-15	100	0.0932
				FA	3.0054	3.0054	3.0054	3.11e-10	100	22.7230
				WPA	3.0054	3.0054	3.0054	3.58e-15	100	0.1742
Ackley	$f_{\min}(\vec{x}) = 0$	50	MIN	GA	11.4570	12.6095	12.1612	0.2719	0	>1.04e+4
				PSO	0.0469	1.7401	0.6846	0.6344	0	>192.5522
				AFSA	20.1600	20.6009	20.4229	0.1009	0	>9.80e+3
				ABC	20.0085	20.0025	20.0061	0.0014	0	>596.3841
				FA	0.0101	0.0209	0.0160	0.0021	0	>4.28e+3
				WPA	8.88e-16	4.44e-15	1.10e-15	8.52e-16	100	7.9476

TABLE 10: Continued.

Function	Global extremum	D	C	Algorithms	Best	Worst	Mean	StdDev	SR/%	Art/s
Griewank	$f_{\min}(\vec{x}) = 0$	100	MN	GA	317.4525	399.6376	363.4174	17.2922	0	$>2.07e + 4$
				PSO	0.0029	0.0082	0.0052	0.0011	0	$>367.0080$
				AFSA	$2.05e + 3$	$2.55e + 3$	$2.33e + 3$	109.6821	0	$>6.51e + 3$
				ABC	$8.95e - 7$	0.0043	$2.26e - 4$	$7.81e - 4$	2	$>620.9561$
				EA	0.0068	0.0118	0.0091	0.0011	0	$>5.72e + 3$
				WPA	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>100</b>	<b>14.5338</b>

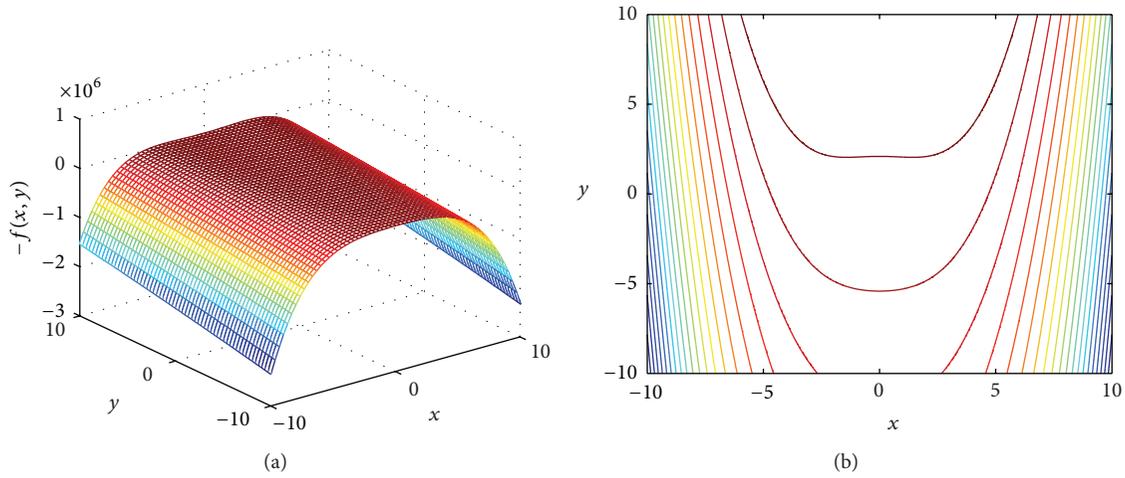


FIGURE 3: Colville function ( $x_1 = x_3, x_2 = x_4$ ): (a) surface plot and (b) contour lines.

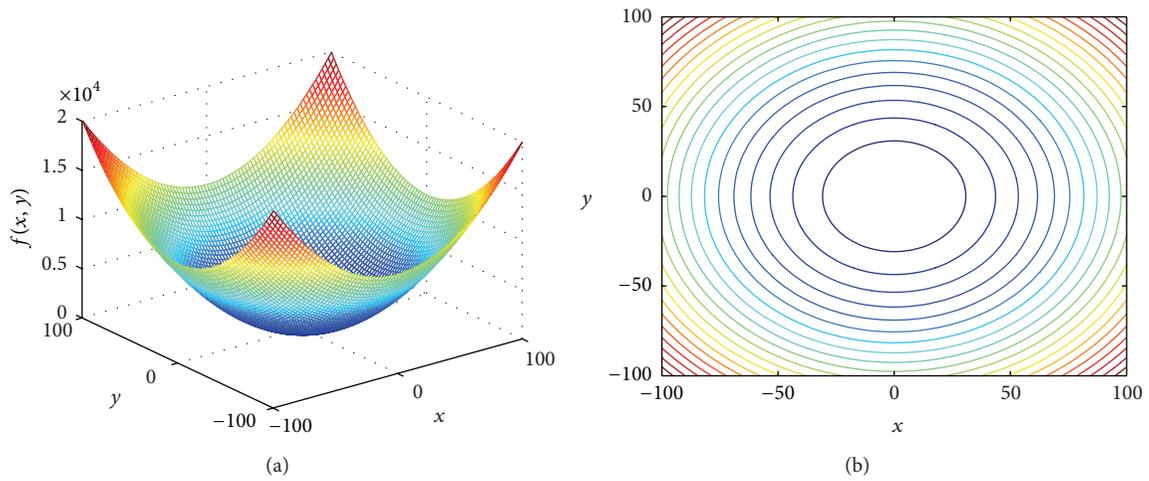


FIGURE 4: Sphere function ( $D = 2$ ): (a) surface plot and (b) contour lines.

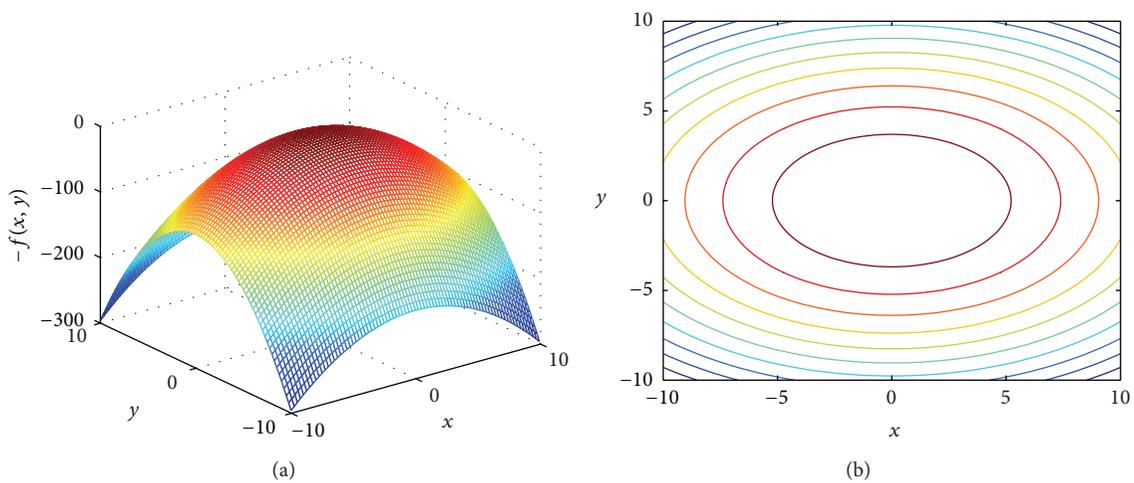


FIGURE 5: Sumsquares function ( $D = 2$ ): (a) surface plot and (b) contour lines.

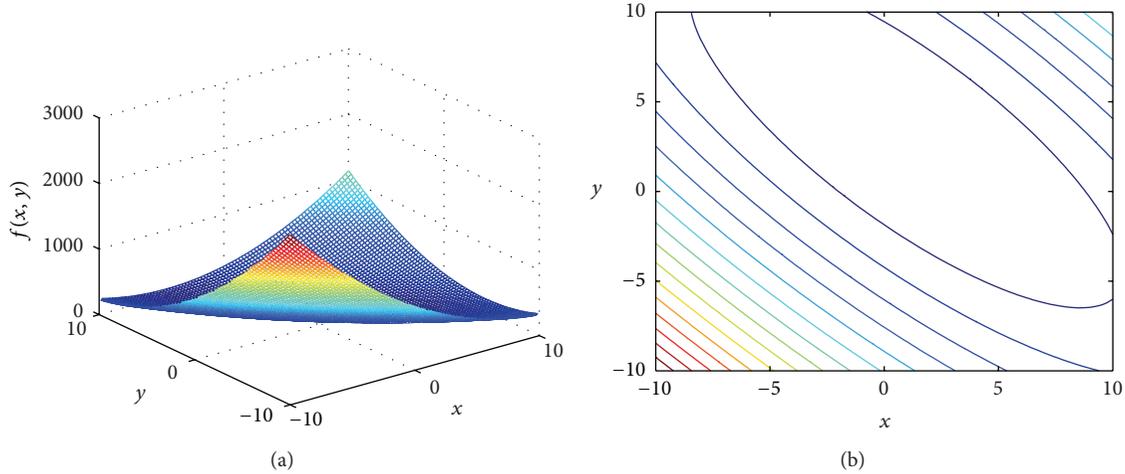


FIGURE 6: Booth function ( $D = 2$ ): (a) surface plot and (b) contour lines.

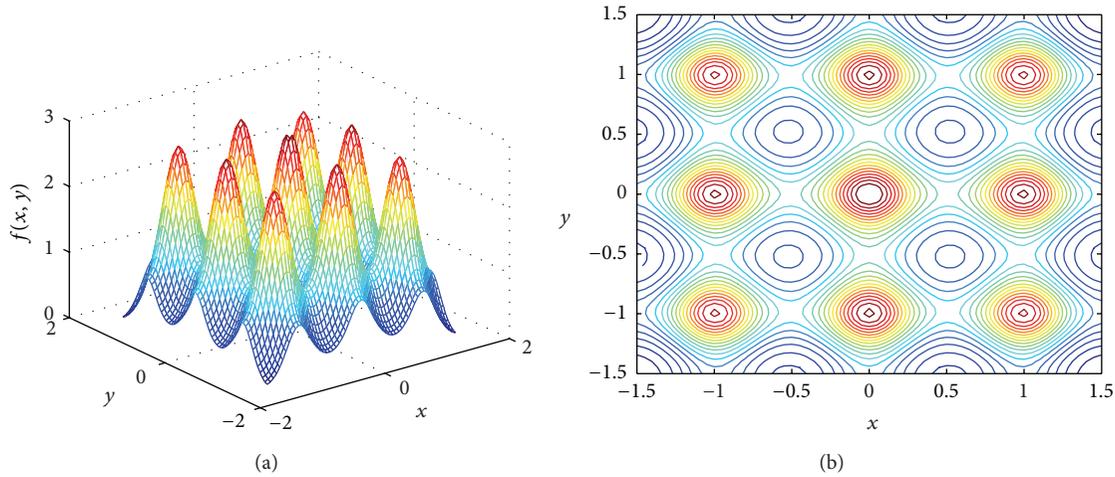


FIGURE 7: Bridge function ( $D = 2$ ): (a) surface plot and (b) contour lines.

Sumsquares functions, when compared with the best results of the other algorithms.

Booth is a multimodal and separable function. Its global minimum value is 0 and optimum solution is  $(x_1, x_2) = (1, 3)$ . When handling Booth function, ABC can get the closer-to-optimal solution within shorter time. Surface plot and contour lines of Booth are shown in Figure 6.

As shown in Figure 6, Booth function has flat surfaces and is difficult for algorithms since the flatness of the function does not give the algorithm any information to direct the search process towards the minima. So WPA does not get the best value as good as ABC, but it can also find good solution and achieve 100% success rate.

Bridge and Ackley are multimodal and nonseparable functions. The global maximum value of Bridge function is 3.0054 and optimum solution is  $(x_1, x_2) \rightarrow (0, 0)$ . The global minimum value of Ackley function is 0 and optimum solution is  $(x_1, x_2, \dots, x_m) = (0, 0, \dots, 0)$ . Surface plot and contour lines of them are separately shown in Figures 7 and 8.

As seen in Figures 7 and 8, the locations of the extremum are regularly distributed, and there are many local extremums near the global extremum. The difficult part of finding optima is that algorithms may easily be trapped in local optima on their way towards the global optimum or oscillate between these local extremums. From Table 10, all algorithms except ASFA show equal performance and achieve 100% success rate on Bridge function. While with respect to Ackley ( $D = 50$ ), only WPA achieves 100% success rate and improves the precision by 13 or 15 orders of magnitude when compared with the best results of other algorithms.

Otherwise, the dimensionality and size of the search space are important issues in the problem [18]. Griewank function, an multimodal and nonseparable function, has the global minimum value of 0 and its corresponding global optimum solution is  $(x_1, x_2, \dots, x_m) = (0, 0, \dots, 0)$ . Moreover, the increment in the dimension of function increases the difficulty. Since the number of local optima increases with the dimensionality, the function is strongly multimodal. Surface

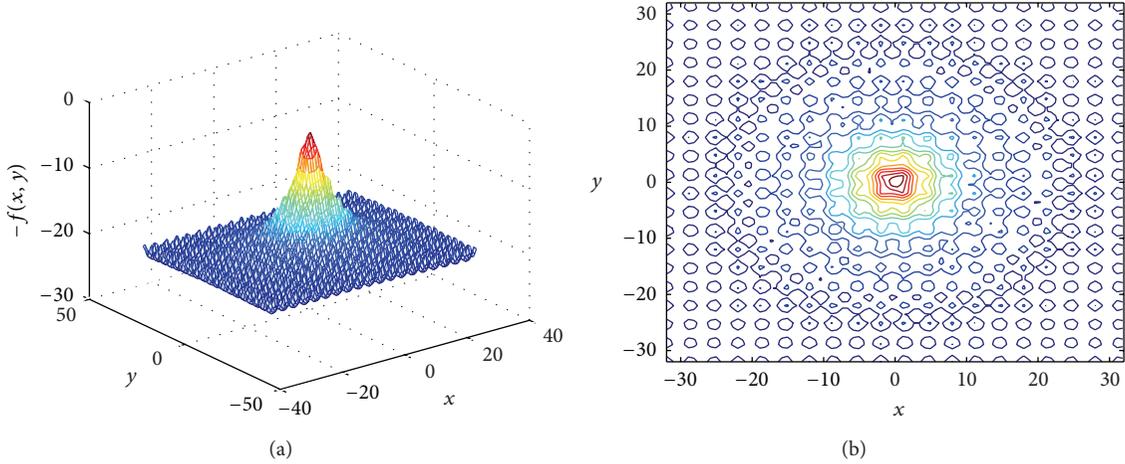


FIGURE 8: Ackley function ( $D = 2$ ): (a) surface plot and (b) contour lines.

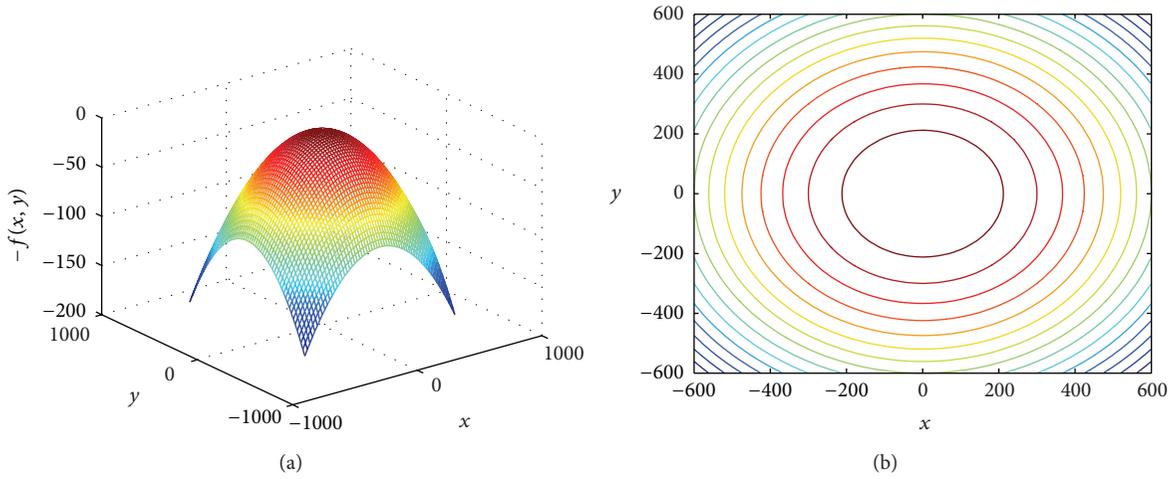


FIGURE 9: Griewank function ( $D = 2$ ): (a) surface plot and (b) contour lines.

plot and contour lines of Griewank function are shown in Figure 9.

WPA with optimized coefficients has good performance in high-dimensional functions. Griewank function ( $D = 100$ ) is a good example. In such a great search space, as shown in Table 10, other algorithms present serious flaws such as premature convergence and difficulty to overcome local minima, while WPA successfully gets the global optimum 0 in 50 runs computation.

As is shown in Table 10, SR shows the robustness of every algorithm, and it means how consistently the algorithm achieves the threshold during all runs performed in the experiments. WPA achieves 100% success rate for functions with different characteristics, which shows its good robustness.

In the experiments, there are 8 functions with variables ranging from 2 to 200. WPA statistically outperforms GA on 6, PSO on 5, ASEFA on 6, ABC on 6, and FA on 7 of these 8 functions. Six of the functions on which GA and ABC are unsuccessful are two unimodal nonseparable functions

(Rosenbrock and Colville) and four high-dimensional functions (Sphere, Sumsquares, Ackley, and Griewank). PSO and FA are unsuccessful on 1 unimodal nonseparable and four high-dimensional functions. But WPA is also not perfect enough for all functions; there are many problems that need to be solved for this new algorithm. From Table 10, on the Rosenbrock function, the accuracy and convergence speed obtained by WPA are not the best ones. So ameliorating WPA inspired by intelligent behaviors of wolves for these special problems is one of our future works. However, so far, it seems to be difficult to simultaneously achieve both fast convergence speed and avoiding local optima for every complex function [19].

It can be drawn that the efficiency of WPA becomes much clearer as the number of variables increases. WPA performs statistically better than the five other state-of-the-art algorithms on high-dimensional functions. Nowadays, high-dimensional problems have been a focus in evolutionary computing domain, since many recent real-world problems (biocomputing, data mining, design, etc.) involve

optimization of a large number of variables [20]. It is convincing that WPA has extensive application in science research and engineering practices.

## 5. Conclusions

Inspired by the intelligent behaviors of wolves, a new swarm intelligent optimization method, wolf pack algorithm (WPA), is presented for locating the global optima of continuous unconstrained optimization problems. We testify the performance of WPA on a suite of benchmark functions with different characteristics and analyze the effect of distance measurements and parameters on WPA. Compared with PSO, ASFA, GA, ABC, and FA, WPA is observed to perform equally or potentially more powerful. Especially for high-dimensional functions such as Sphere ( $D = 200$ ), Sumsquares ( $D = 150$ ), Ackley ( $D = 50$ ), and Griewank ( $D = 100$ ), WPA may be a better choice, since WPA possesses superior performance in terms of accuracy, convergence speed, stability, and robustness.

After all, WPA is a new attempt and achieves some success for global optimization, which can provide new ideas for solving engineering and science optimization problems. In future, different improvements can be made on the WPA algorithm and tests can be made on more different test functions. Meanwhile, practical applications in areas of classification, parameters optimization, engineering process control, and design and optimization of controller would also be worth further studying.

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## References

- [1] F. Kang, J. Li, and Z. Ma, "Rosenbrock artificial bee colony algorithm for accurate global optimization of numerical functions," *Information Sciences*, vol. 181, no. 16, pp. 3508–3531, 2011.
- [2] C. Grosan and A. Abraham, "A novel global optimization technique for high dimensional functions," *International Journal of Intelligent Systems*, vol. 24, no. 4, pp. 421–440, 2009.
- [3] Y. Yang, Y. Wang, X. Yuan, and F. Yin, "Hybrid chaos optimization algorithm with artificial emotion," *Applied Mathematics and Computation*, vol. 218, no. 11, pp. 6585–6611, 2012.
- [4] W. S. Gao and C. Shao, "Pseudo-collision in swarm optimization algorithm and solution: rain forest algorithm," *Acta Physica Sinica*, vol. 62, no. 19, Article ID 190202, pp. 1–15, 2013.
- [5] Y. Celik and E. Ulker, "An improved marriage in honey bees optimization algorithm for single objective unconstrained optimization," *The Scientific World Journal*, vol. 2013, Article ID 370172, 11 pages, 2013.
- [6] E. Cuevas, D. Zaldivar, and M. Pérez-Cisneros, "A swarm optimization algorithm for multimodal functions and its application in multicircle detection," *Mathematical Problems in Engineering*, vol. 2013, Article ID 948303, 22 pages, 2013.
- [7] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, pp. 1942–1948, Perth, Australia, December 1995.
- [8] M. Dorigo, *Optimization, learning and natural algorithms [Ph.D. thesis]*, Politecnico di Milano, Milano, Italy, 1992.
- [9] X.-L. Li, Z.-J. Shao, and J.-X. Qian, "Optimizing method based on autonomous animats: Fish-swarm Algorithm," *System Engineering Theory and Practice*, vol. 22, no. 11, pp. 32–38, 2002.
- [10] D. Karaboga, "An idea based on honeybee swarm for numerical optimization," Tech. Rep. TR06, Computer Engineering Department, Engineering Faculty, Erciyes University, Kayseri, Turkey, 2005.
- [11] X.-S. Yang, "Firefly algorithms for multimodal optimization," in *Stochastic Algorithms: Foundations and Applications*, vol. 5792 of *Lecture Notes in Computer Science*, pp. 169–178, Springer, Berlin, Germany, 2009.
- [12] J. A. Ruiz-Vanoye, O. Díaz-Parra, F. Cocón et al., "Meta-Heuristics algorithms based on the grouping of animals by social behavior for the travelling sales problems," *International Journal of Combinatorial Optimization Problems and Informatics*, vol. 3, no. 3, pp. 104–123, 2012.
- [13] C.-G. Liu, X.-H. Yan, and C.-Y. Liu, "The wolf colony algorithm and its application," *Chinese Journal of Electronics*, vol. 20, no. 2, pp. 212–216, 2011.
- [14] D. E. Goldberg, *Genetic Algorithms in Search, Optimisation and Machine Learning*, Addison-Wesley, Reading, Mass, USA, 1989.
- [15] S.-K. S. Fan and E. Zahara, "A hybrid simplex search and particle swarm optimization for unconstrained optimization," *European Journal of Operational Research*, vol. 181, no. 2, pp. 527–548, 2007.
- [16] P. Caamaño, F. Bellas, J. A. Becerra, and R. J. Duro, "Evolutionary algorithm characterization in real parameter optimization problems," *Applied Soft Computing*, vol. 13, no. 4, pp. 1902–1921, 2013.
- [17] D. Ortiz-Boyer, C. Hervás-Martínez, and N. García-Pedrajas, "CIXL2: a crossover operator for evolutionary algorithms based on population features," *Journal of Artificial Intelligence Research*, vol. 24, pp. 1–48, 2005.
- [18] M. S. Kuran and M. Gündüz, "A recombination-based hybridization of particle swarm optimization and artificial bee colony algorithm for continuous optimization problems," *Applied Soft Computing*, vol. 13, no. 4, pp. 2188–2203, 2013.
- [19] W. Gao and S. Liu, "Improved artificial bee colony algorithm for global optimization," *Information Processing Letters*, vol. 111, no. 17, pp. 871–882, 2011.
- [20] Y. F. Ren and Y. Wu, "An efficient algorithm for high-dimensional function optimization," *Soft Computing*, vol. 17, no. 6, pp. 995–1004, 2013.

## Research Article

# Research on Cooperative Combat for Integrated Reconnaissance-Attack-BDA of Group LAVs

Li Bing, Li Jie, He Guanglin, and Li Dalin

*School of Mechatronic Engineering, Beijing Institute of Technology, Beijing 100081, China*

Correspondence should be addressed to He Guanglin; [heguanglinbit@163.com](mailto:heguanglinbit@163.com)

Received 31 August 2013; Revised 3 January 2014; Accepted 18 January 2014; Published 9 March 2014

Academic Editor: Vishal Bhatnagar

Copyright © 2014 Li Bing et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

LAVs (loitering air vehicles) are advanced weapon systems that can loiter autonomously over a target area, detect and acquire the targets, and then attack them. In this paper, by the theory of Itô stochastic differential, a group system was analyzed. The uniqueness and continuity of the solution of the system was discussed. Afterwards the model of the system based on the state transition was established with the finite state machine automatically. At last, a search algorithm was proposed for obtaining good feasible solutions for problems. And simulation results show that model and method are effective for dealing with cooperative combat of group LAVs.

## 1. Introduction

LAV is a new kind of aerial vehicle [1], which can loiter in the air over the targets. It comes out with the development of UAV and munitions. While a single LAV performing a single task will bring some benefits, greater benefits will come from the cooperation of group LAVs. The main motivation for group cooperation stems from the possible synergy, as the group performance is exceed the sum of the performance of the individual LAVs [2].

The recent development of group LAV technology has a great number of interests within the intelligence gathering and remote sensing communities [3]. Through cooperation, the team can reconfigure its distribution architecture to minimize the performance degradation to such expected failures. Such cooperation should take advantage of the following capabilities available to the group: Global Information, Resource Management, and Robustness.

In the past few years, the problem of cooperation problems has been widely studied by researchers. Group cooperation is the strongest degree of cohesive group action. Consider a set of LAVs that have been designated to be part of the group; the group could have more than one payoff function that it wishes to optimize, which would then entail multiobjective optimization [4].

However, the current study, whether for its cooperation with automatic organization or conscious cooperation, is

carried out only for a specific problem. Flight formation is investigated in [5–7]. In [8], the method of Mixed Integer Linear Programming approach is applied to path planning. A similar cooperative path planning problem is also discussed in [9–11]. Cooperative task allocation problem is discussed in [12, 13] using game theory. In [14, 15], cooperative task allocation problem is discussed by intelligent algorithm. Collision and obstacle avoidance are discussed in [16, 17]. Cooperative information consensus is discussed in [18–20] and many others [21, 22].

In this paper we focus on autonomous LAVs, which are designed to operate as a pack of vehicles that search, detect, and attack targets. We develop analytic probability model for analyzing some design and operational aspects relating to group LAVs. The rest of the paper is organized as follows. We describe the differential equation model with finite state machine automatically. And the optimization function is established based on the combat efficiency. Finally, we provide some test results and give our conclusions.

## 2. Problem Formulation and Analysis

*2.1. The Behaviors in the Process of Cooperative Combat.* Each LAV has an identical set of behaviors governed by the same controller. A LAV can be in one of four possible situations: Searching, Attacking, BDA, and Removed.

The rules of the behavior are as in Algorithm 1.

**If** the LAV satisfied the physical characteristics and communication capability,  
**then** over the target area searching for valuable targets;  
**If** the LAV confirms the targets are the valuable, **then** acquires the target and attacks it;  
**If** After the attacks on targets, **then** over the target area for BDA;  
**If** achieve the targets' BDA, **then** enter the next round of decisions;

ALGORITHM 1

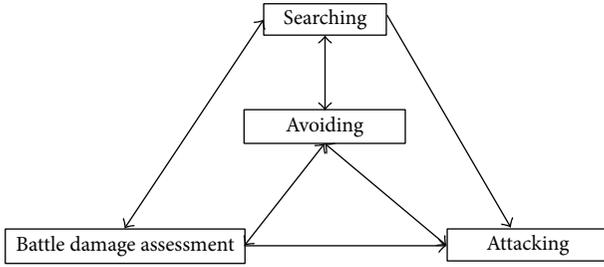


FIGURE 1: State diagram of the group LAVs behaviors.

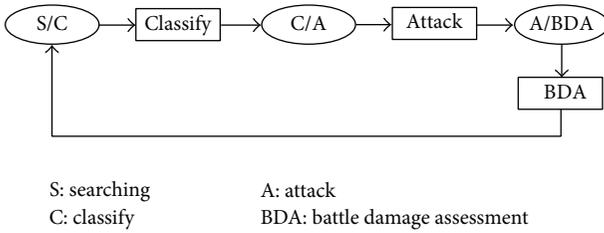


FIGURE 2: LAV mission timeline.

Figure 1 shows the sequence of behaviors that the LAV engages in during the combat process. Figure 2 describes the aforementioned mission time parameters.

**2.2. Influence Analysis of the Group LAVs System.** Consider dynamic model described by the following equation;

$$F_i(t) = \left( \prod_{j=1}^N U_{ji}(t)^{\alpha_{ji}} \right) S_i(t)^{\beta_i} E_i(t)^{\gamma_i}, \quad (1)$$

$$i = 1, 2, \dots, N; \quad j = 1, 2, \dots, N, \quad j \neq i,$$

where  $U_{ji}(t)$  is the influence from  $j$  individual of Swarms system,  $S_i(t)$  is the UAV  $i$  self-influence,  $E_i$  is the positive factor of environment, and  $\alpha_{i,j}$  and  $\beta_i$  are the constant parameters.

In order to describe the influence function from the system science we use the *Itô* stochastic differential.

Set

$$U_i(t)^{\alpha_i} = \prod_{j=1}^N U_{ji}(t)^{\alpha_{ji}}, \quad (2)$$

satisfying  $\beta_i + \alpha_i = 1$ ,  $i = 1, \dots, N$ .

According to the *Itô* stochastic differential we can get the stochastic differential equations of the factor function  $F_i(t)$ :

$$\begin{aligned} dF_i(t) = & \frac{\partial F_i(t)}{\partial (U_i, S_i, E_i, t)} + \frac{1}{2} \frac{\partial^2 F_i(t)}{\partial (U_i^2, E_i^2, S_i^2)} \\ & + \frac{\partial^2 F_i(t)}{\partial U_i \partial S_i} dU_i(t) dS_i(t) + \frac{\partial^2 F_i(t)}{\partial U_i \partial E_i} dU_i dE_i \quad (3) \\ & + \frac{\partial^2 F_i(t)}{\partial S_i \partial E_i} dS_i dE_i. \end{aligned}$$

**Theorem 1.** Consider the following *Itô* stochastic differential equation:

$$\begin{aligned} dF_i(t) = & \frac{\partial F_i(t)}{\partial (U_i, S_i, E_i, t)} + \frac{1}{2} \frac{\partial^2 F_i(t)}{\partial (U_i^2, E_i^2, S_i^2)} \\ & + \frac{\partial^2 F_i(t)}{\partial U_i \partial S_i} dU_i(t) dS_i(t) + \frac{\partial^2 F_i(t)}{\partial U_i \partial E_i} dU_i dE_i \quad (4) \\ & + \frac{\partial^2 F_i(t)}{\partial S_i \partial E_i} dS_i dE_i. \end{aligned}$$

Structure a Borel measurable function and satisfied Lipschitz condition. Then if

$$F_i(x, y, z, 0) = \text{Const}, \quad E(F_i(x, y, z, 0))^2 < +\infty, \quad (5)$$

the differential equation  $dF_i(t)$  has the unique and continuous solution  $F_i(t)$  and convergence in probability.

*Proof.* The stochastic differential equation is derived by Variable replacement, which is described as follows:

$$\begin{aligned} F_{U_i U_i} &= \alpha_i (\alpha_i - 1) \rho^2 U_i^{\alpha_i} S_i^{\beta_i} E_i^{\gamma_i} dt, \\ F_{S_i S_i} &= \beta_i (\beta_i - 1) \sigma^2 U_i^{\alpha_i} S_i^{\beta_i} E_i^{\gamma_i} dt, \\ F_{E_i E_i} &= \gamma_i (\gamma_i - 1) \delta^2 U_i^{\alpha_i} S_i^{\beta_i} E_i^{\gamma_i} dt, \\ F_{U_i S_i} &= \alpha_i \beta_i \sigma \rho U_i^{\alpha_i} S_i^{\beta_i} E_i^{\gamma_i} dt, \\ F_{U_i E_i} &= \alpha_i \gamma_i \rho \delta U_i^{\alpha_i} S_i^{\beta_i} E_i^{\gamma_i} dt, \\ F_{S_i E_i} &= \alpha_i \gamma_i \sigma \delta U_i^{\alpha_i} S_i^{\beta_i} E_i^{\gamma_i} dt. \end{aligned} \quad (6)$$

Then the equation  $dF_i(t)$  converts into the following form:

$$dF_i(t) = \left( \begin{array}{c} \frac{1}{2} (\alpha_i (\alpha_i - 1) \rho^2 + \beta_i (\beta_i - 1) \sigma^2 + \gamma_i (\gamma_i - 1) \delta^2) \\ (\alpha_i \rho + \beta_i \sigma + \gamma_i \delta) \end{array} \right) + \left[ \begin{array}{c} (\alpha_i \beta_i \sigma \rho + \alpha_i \gamma_i \rho \delta + \beta_i \gamma_i \sigma \delta) \\ 0 \end{array} \right]^T \left[ \begin{array}{c} dt \\ dz \end{array} \right] F_i(t). \quad (7)$$

So we structure the Borel measurable function as follows:

$$g(t, F_i) = \frac{1}{2} ((\alpha_i (\alpha_i - 1) \rho^2 + \beta_i (\beta_i - 1) \sigma^2 + \gamma_i (\gamma_i - 1) \delta^2) + 2(\alpha_i \beta_i \sigma \rho + \alpha_i \gamma_i \rho \delta + \beta_i \gamma_i \sigma \delta + n\rho + s\alpha + \delta m)) \times F_i(t) h(t, F_i) = (\alpha_i \rho + \beta_i \sigma + \gamma_i \delta) F_i(t). \quad (8)$$

According to the lemma in [23] lemma, we will prove that the Borel measurable function satisfied the Lipchitz condition:

$$\begin{aligned} & |g(t, F_{i1}) - f(t, F_{i2})| + |h(t, F_{i1}) - \delta(t, F_{i2})| \\ &= \left| \frac{1}{2} (\alpha_i (\alpha_i - 1) \rho^2 + \beta_i (\beta_i - 1) \sigma^2 + \gamma_i (\gamma_i - 1) \delta^2) F_i(t) \right| \\ & \quad \times |F_{i1} - F_{i2}| + |\alpha_i \beta_i \sigma \rho + \alpha_i \gamma_i \rho \delta + \beta_i \gamma_i \sigma \delta + n\rho + s\alpha + \delta m| \\ & \quad \times F_i(t) |F_{i1} - F_{i2}| + (\alpha_i \rho + \beta_i \sigma + \gamma_i \delta) |F_{i1} - F_{i2}| \\ &\leq \left| \frac{1}{2} (\alpha_i (\alpha_i - 1) \rho^2 + \beta_i (\beta_i - 1) \sigma^2 + \gamma_i (\gamma_i - 1) \delta^2) \right| \\ & \quad \times |F_{i1} - F_{i2}| + |(\alpha_i \beta_i \sigma \rho + \alpha_i \gamma_i \rho \delta + \beta_i \gamma_i \sigma \delta)| |F_{i1} - F_{i2}| \\ & \quad + |(n\rho + s\alpha + \delta m + \alpha_i \rho + \beta_i \sigma + \gamma_i \delta)| |F_{i1} - F_{i2}|, \\ & |g(t, F_i)|^2 + |h(t, F_i)|^2 \\ &= \frac{1}{4} (\alpha_i (\alpha_i - 1) \rho^2 + \beta_i (\beta_i - 1) \sigma^2 + \gamma_i (\gamma_i - 1) \delta^2) \\ & \quad + 2\alpha_i \beta_i \sigma \rho + 2\alpha_i \gamma_i \rho \delta + 2\beta_i \gamma_i \sigma \delta + 2n\rho + 2s\alpha)^2 F_i^2 \\ & \quad + (\alpha_i \rho + \beta_i \sigma + \gamma_i \delta)^2 F_i^2 \\ &\leq \left\{ [\alpha_i (\alpha_i - 1) \rho^2 + \beta_i (\beta_i - 1) \sigma^2 + \gamma_i (\gamma_i - 1) \delta^2 \right. \\ & \quad \left. + (\alpha_i \beta_i \sigma \rho + \alpha_i \gamma_i \rho \delta + \beta_i \gamma_i \sigma \delta + n\rho + s\alpha + \delta m)]^2 \right. \\ & \quad \left. + (\alpha_i \rho + \beta_i \sigma + \gamma_i \delta)^2 \right\} F_i^2, \\ \omega_1 &= \left| \frac{1}{2} (\alpha_i (\alpha_i - 1) \rho^2 + \beta_i (\beta_i - 1) \sigma^2 + \gamma_i (\gamma_i - 1) \delta^2) \right| \\ & \quad + |(\alpha_i \beta_i \sigma \rho + \alpha_i \gamma_i \rho \delta + \beta_i \gamma_i \sigma \delta)| \\ & \quad + |(n\rho + s\alpha + \delta m + \alpha_i \rho + \beta_i \sigma + \gamma_i \delta)|, \end{aligned}$$

$$\begin{aligned} \omega_2 &= \sqrt{2} * \text{Max} \left\{ [\alpha_i (\alpha_i - 1) \rho^2 + \beta_i (\beta_i - 1) \sigma^2 + \gamma_i (\gamma_i - 1) \delta^2 \right. \\ & \quad \left. + (\alpha_i \beta_i \sigma \rho + \alpha_i \gamma_i \rho \delta + \beta_i \gamma_i \sigma \delta + n\rho \right. \\ & \quad \left. + s\alpha + \delta m)]^2, (\alpha_i \rho + \beta_i \sigma + \gamma_i \delta)^2 \right\}, \\ \omega &= \text{Max} \{ \omega_1, \omega_2 \}. \quad (9) \end{aligned}$$

So we can get the equation and satisfy the Lipchitz condition as follows:

$$\begin{aligned} & |g(t, F_{i1}) - h(t, F_{i2})| + |\sigma(t, F_{i1}) - \sigma(t, F_{i2})| \leq \omega |F_{i1} - F_{i2}|, \\ & |g(t, F_i)|^2 + |h(t, F_i)|^2 \leq \omega^2 (1 + |F_i|^2). \quad (10) \end{aligned}$$

The conclusion that the influence function  $F_i(t)$  has unique and continuous solution and convergence in probability 1 is correct.  $\square$

### 3. Mathematical Model and Analysis of System

*3.1. Mathematical Model of the System Based on the State Transition.* Differential equation model was established using finite state machine automatically:

$$\begin{aligned} \frac{dN_{\text{Searching}}(t)}{dt} &= \frac{1}{\lambda_{\text{Avoiding}}} N_{\text{AS}}(t) + \frac{1}{\lambda_{\text{BDA}}} N_{\text{BDA}}(t) \\ & \quad - \alpha_s N_{\text{Searching}}(t) N_{\text{Target}}(t) - \alpha_{\text{AS}} N_{\text{Searching}}(t) \\ & \quad \times (N_{\text{Searching}}(t) + N + B), \\ \frac{dN_{\text{AS}}(t)}{dt} &= -\frac{1}{\lambda_{\text{Avoiding}}} N_{\text{AS}}(t) + \alpha_{\text{AS}} N_{\text{Searching}}(t) \\ & \quad \times (N_{\text{Searching}}(t) + N + B), \\ \frac{dN_{\text{AA}}(t)}{dt} &= -\frac{1}{\lambda_{\text{Avoiding}}} N_{\text{AA}}(t) + \alpha_{\text{AA}} N_{\text{Attacking}}(t) \\ & \quad \times (N_{\text{Attacking}}(t) + N + B), \\ \frac{dN_{\text{AB}}(t)}{dt} &= -\frac{1}{\lambda_{\text{Avoiding}}} N_{\text{AB}}(t) + \alpha_{\text{AB}} N_{\text{BDA}}(t) \\ & \quad \times (N_{\text{BDA}}(t) + N + B), \\ \frac{dN_{\text{Attacking}}(t)}{dt} &= \alpha_s N_{\text{Searching}}(t) N_{\text{Target}}(t) \\ & \quad - \frac{1}{\lambda_{\text{Attacking}}} N_{\text{Attacking}}(t), \end{aligned}$$

$$\begin{aligned} \frac{dN_{\text{BDA}}(t)}{dt} &= \alpha_{\text{AB}} N_{\text{AB}}(t) - \frac{1}{\lambda_{\text{BDA}}} N_{\text{BDA}}(t) + \alpha_{\text{AB}} N_{\text{BDA}}(t) \\ &\quad \times (N_{\text{BDA}}(t) + N + B), \\ \frac{dN_{\text{Target}}(t)}{dt} &= -\alpha_s N_{\text{Searching}}(t) N_{\text{Target}}(t). \end{aligned} \quad (11)$$

*Parameters Notation.*  $N$  is the number of LAVs in the system,  $N_{\text{Searching}}(t)$  is the number of LAVs in the searching state at time  $t$ ,  $N_{\text{Avoiding}}(t)$  is the number of LAVs in the avoiding state at time  $t$ ,  $N_{\text{Attacking}}(t)$  is the number of LAVs in the attacking state at time  $t$ ,  $N_{\text{BDA}}(t)$  is the number of LAVs in the BDA state at time  $t$ ,  $N_{\text{Target}}(t)$  is the number of targets at time  $t$ ,  $\lambda_{\text{Avoiding}}$ ,  $\lambda_{\text{Attacking}}$ , and  $\lambda_{\text{BDA}}$  are the average sustained time of Avoiding, Attacking, and BDA,  $\alpha_s$  is the rate of searching a valuable target, and  $\alpha_{\text{AS}}$ ,  $\alpha_{\text{AA}}$ ,  $\alpha_{\text{AB}}$  are the rate of encounter obstacles on Searching, Attacking, and BDA.

*3.2. Cost Function Based on Effectiveness.* In this paper we consider the total flight distance of the group system:

$$\min \sum_{k=1}^K \left( \sum_{l=1}^{N_c} \sum_{i=1}^{N_u} \sum_{j=1}^{N_{\text{target}}} C_{l,i,j}^{X_{l-1}} x_{l,i,j} \right). \quad (12)$$

Parameter  $x_{l,i,j} \in \{0, 1\}$  is the decision variable and  $s \in \{1, 2, 3\}$  is the stages of the task; if  $l \in s$ , then  $x_{l,i,j} = 1$ ; else  $x_{l,i,j} = 0$ , and  $c_{l,i,j}^{X_{l-1}}$  is the distance for LAV  $i$  to the target  $j$ .

Subject to

$$\begin{aligned} \sum_{l=1}^{N_c} \sum_{i=1}^{N_u} x_{l,i,j} &= N, \quad j \in T, \\ \sum_{l=1}^{N_c} \sum_{j=1}^{N_{\text{target}}} r_{l,i,j}^{X_{l-1}} x_{l,i,j} &\leq b_i, \quad i \in U, \\ d_{A \min} &\geq d_{\min} + v \Delta t_1, \\ d_{V \min} &\geq d_{V \min} + v \Delta t_2. \end{aligned} \quad (13)$$

Parameter  $b_i$  is the flight capability and  $\Delta t_m$  is the min. time between the tasks.

*3.3. Search Algorithm.* According to the proposed cooperative combat, each LAV uses a cost function to select and update its behaviour. This method is quite flexible in that it allows the characterization of various mission-level objectives.

The flowchart depicting the optimization algorithm for the LAV group scenario is shown in Figure 3.

The data as in Tables 1 and 2.

In this example, eight LAVs are searching an area containing three targets of similar types.

LAVs trajectories through the 10 seconds are shown in Figure 4; at  $t = 10$  s the targets are discovered by the searching

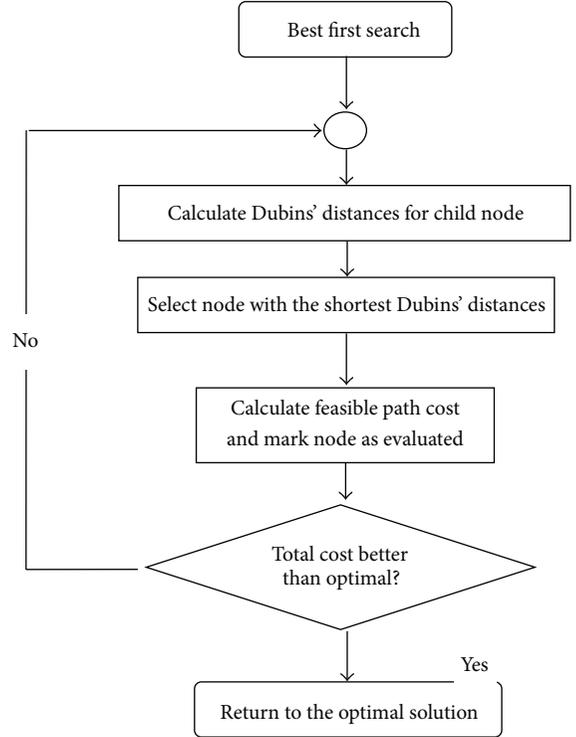


FIGURE 3: Search flowchart.

TABLE 1: Dynamics performance parameters.

Parameter	Value
Loitering velocity	100 m/s
Maximum range	180 km
Minimum turning radius	500 m

TABLE 2: Initialization of the MAV.

Index	Location (km)	Orientation (rad)	Period of flight time(s)
LAV1	(3.93, 18.34)	3.335	129
LAV2	(5.02, 5.72)	4.895	635
LAV3	(12.32, 15.14)	5.868	248
LAV4	(9.47, 15.07)	0.816	422
LAV5	(7.03, 7.61)	3.574	132
LAV6	(16.62, 11.36)	2.949	481
LAV7	(11.71, 1.52)	0.074	210
LAV8	(10.99, 1.08)	2.118	523

LAVs, nearly simultaneously. LAV1, 5 and 6 are assigned to verify the targets.

LAVs trajectories through the 50 seconds are shown in Figure 5, at  $t = 39$  s Target 3 is classified by LAV6, at  $t = 44$  s Target 1 is classified by LAV1, at  $t = 49$  s Target 2 is classified by LAV5, and then the LAVs perform attack on targets.

LAVs trajectories through the 115 seconds are shown in Figure 6, at  $t = 91$  s Targets 1 was attacked by LAV4, then LAV2 performs BDA on target 1.

LAVs trajectories through the 235 seconds are shown in Figure 7, at  $t = 120$  s Targets 2 was attacked by LAV8, then

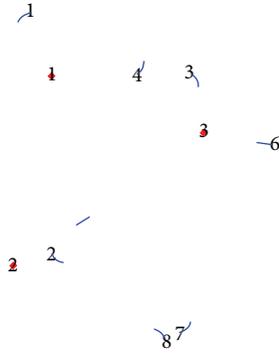


FIGURE 4:  $T = 10$  s trajectories with respect to time.

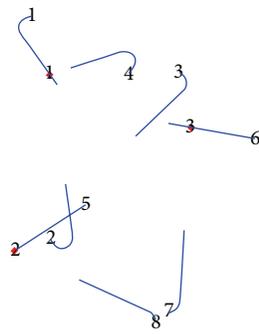


FIGURE 5:  $T = 50$  s trajectories with respect to time.

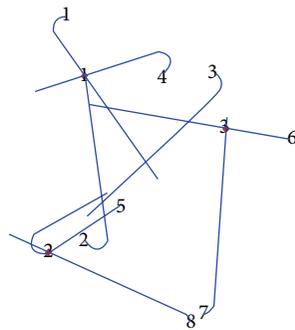


FIGURE 6:  $T = 115$  s trajectories with respect to time.

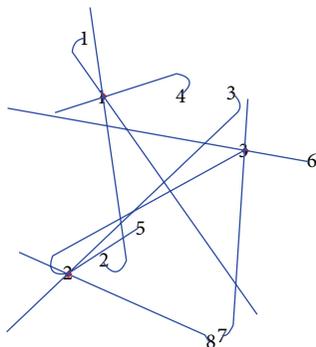


FIGURE 7:  $T = 235$  s trajectories with respect to time.

LAV3 performs BDA on target 2, at  $t = 138$  s Targets 3 was attacked by LAV7, then LAV5 performs BDA on target 3.

At  $T = 230$  s, the tasks are completely finished by the group LAVs.

#### 4. Conclusion

In this paper the problem of the cooperative combat associated with the group LAVs system has been solved. Afterwards, by the theory of *Itô* stochastic differential, a group system was analyzed. The model of the system based on the state transition was established with the finite state machine automatically. At last, a search algorithm was proposed for obtaining good feasible solutions for problems. And we took simulation tests to verify the conclusion.

#### Conflict of Interests

The authors declare that they have no conflict of interests regarding the publication of this paper.

#### Acknowledgment

This paper is supported by the National Defense Pre-Research Foundation of China (Grant no. B222011XXXX).

#### References

- [1] D. Li, J. Li, Z. Guan, and J. Wang, "Improved GA based multi-LAVs' aerial delivery areas-choosing method," *Applied Mechanics and Materials*, vol. 157-158, pp. 1498–1505, 2012.
- [2] T. Shima and S. Rasmussen, *UAV Cooperative Decision and Control Challenges and Practical Approaches*, Society for Industrial Mathematics, 2009.
- [3] F. Namin, J. S. Petko, and D. H. Werner, "Analysis and design optimization of robust aperiodic Micro-UAV swarm-based antenna arrays," *IEEE Transactions on Antennas and Propagation*, vol. 60, no. 5, pp. 2295–2308, 2012.
- [4] R. Luce and H. Raiffa, *Games and Decisions: Introduction and Critical Survey*, Dover Publications, New York, NY, USA, 1989.
- [5] H. Kim and Y. Kim, "Trajectory optimization for unmanned aerial vehicle formation reconfiguration," *Engineering Optimization*, vol. 46, no. 1, pp. 84–106, 2014.
- [6] I. Bayezit and B. Fidan, "Distributed cohesive motion control of flight vehicle formations," *IEEE Transactions on Industrial Electronics*, vol. 60, no. 12, pp. 5763–5772, 2013.
- [7] Z. Chao, S. Zhou, L. Ming, and W. Zhang, "UAV formation flight based on nonlinear model predictive control," *Mathematical Problems in Engineering*, vol. 2012, Article ID 261367, 15 pages, 2012.
- [8] J. Tisdale, Z. W. Kim, and J. K. Hedrick, "Autonomous UAV path planning and estimation: an online path planning framework for cooperative search and localization," *IEEE Robotics and Automation Magazine*, vol. 16, no. 2, pp. 35–42, 2009.
- [9] A. T. Klesh, P. T. Kabamba, and A. R. Girard, "Path planning for cooperative time-optimal information collection," in *Proceedings of the American Control Conference (ACC '08)*, pp. 1991–1996, June 2008.

- [10] Y. Kim, D. Gu, and I. Postlethwaite, "Real-time optimal mission scheduling and flight path selection," *IEEE Transactions on Automatic Control*, vol. 52, no. 6, pp. 1119–1123, 2007.
- [11] P. B. Sujit and R. Beard, "Cooperative path planning for multiple UAVs exploring an unknown region," in *Proceedings of the American Control Conference (ACC '07)*, pp. 5446–5451, July 2007.
- [12] H. Chen, D. Shen, G. Chen, E. P. Blasch, and K. Pham, "Tracking evasive objects via a search allocation game," in *Proceedings of the American Control Conference (ACC '10)*, pp. 6981–6986, July 2010.
- [13] Q. Han, X. M. Wang, and J. Dang, "Research on the task allocation model and algorithm of MAS based on game theory," *Computer Applications and Software*, pp. 51–53, 2012.
- [14] M. Darrach, E. Fuller, T. Munasinghe, K. Duling, M. Gautam, and M. Wathen, "Using genetic algorithms for tasking teams of raven UAVs," *Journal of Intelligent and Robotic Systems*, vol. 70, pp. 361–371, 2013.
- [15] M. Yao, X. Wang, and M. Zhao, "Cooperative combat task assignment optimization design for unmanned aerial vehicles cluster," *Journal of University of Electronic Science and Technology of China*, pp. 723–727, 2013.
- [16] R. Olfati-Saber and R. M. Murray, "Mesh stability of unmanned aerial vehicle cluster," in *Proceedings of the American Control Conference (ACC '01)*, pp. 62–68, Arlington, Va, USA, June 2001.
- [17] C. Tomlin, G. J. Pappas, and S. Sastry, "Conflict resolution for air traffic management: a study in multiagent hybrid systems," *IEEE Transactions on Automatic Control*, vol. 43, no. 4, pp. 509–521, 1998.
- [18] L. Bing, L. Jie, and H. KeWei, "Modeling and flocking consensus analysis for large-scale UAV swarms," *Mathematical Problems in Engineering*, vol. 2013, Article ID 368369, 9 pages, 2013.
- [19] R. W. Beard and V. Stepanyan, "Information consensus in distributed multiple vehicle coordinated control," in *Proceedings of the 43th IEEE Conference on Decision and Control*, pp. 2029–2034, Maui, Hawaii, USA, December 2003.
- [20] J. Luo, "Some new optimal control problem in UAV cooperative control with information flow constraints," in *Proceedings of the American Control Conference (ACC'03)*, pp. 2181–2186, Denver, Colo, USA, June 2003.
- [21] B. Li, "Stochastic process model of the multi-UAVs collaborative system based on state transition," in *Proceedings of the 2012 International Conference on Modeling, Identification and Control*, pp. 757–761, 2012.
- [22] D. L. Li, *Group decision making and control method for micro-unmanned aerial vehicle [Ph.D. thesis]*, Beijing Institute of Technology, 2011.
- [23] B. Li, "Analysis of influence of multiple UAVs' coordination system," *Journal of Information and Computational Science*, vol. 10, no. 2, pp. 571–578, 2013.

## Research Article

# Optimum Performance-Based Seismic Design Using a Hybrid Optimization Algorithm

S. Talatahari,<sup>1</sup> A. Hosseini,<sup>2</sup> S. R. Mirghaderi,<sup>2</sup> and F. Rezazadeh<sup>2</sup>

<sup>1</sup> Department of Civil Engineering, University of Tabriz, Tabriz, Iran

<sup>2</sup> Department of Civil Engineering, Faculty of Engineering, University of Tehran, Tehran, Iran

Correspondence should be addressed to S. Talatahari; [siamak.talat@gmail.com](mailto:siamak.talat@gmail.com)

Received 16 May 2013; Accepted 21 November 2013; Published 16 February 2014

Academic Editor: Yudong Zhang

Copyright © 2014 S. Talatahari et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A hybrid optimization method is presented to optimum seismic design of steel frames considering four performance levels. These performance levels are considered to determine the optimum design of structures to reduce the structural cost. A pushover analysis of steel building frameworks subject to equivalent-static earthquake loading is utilized. The algorithm is based on the concepts of the charged system search in which each agent is affected by local and global best positions stored in the charged memory considering the governing laws of electrical physics. Comparison of the results of the hybrid algorithm with those of other metaheuristic algorithms shows the efficiency of the hybrid algorithm.

## 1. Introduction

Many of engineering problems can be modeled into optimization problems. Therefore, developing new optimization techniques to solve these types of problems becomes highly significant. For this case, a large number of optimization algorithms as optimization techniques have already been proposed and applied in solving engineering problems such as ant colony optimization (ACO) [1], ABC (artificial bee colony) [2, 3], cuckoo search (CS) [4–6], bat algorithm (BA) [7, 8], genetic programming (GP) [9], ES (evolutionary strategy) [10], GA (genetic algorithm) [11], HS (harmony search) [12, 13], biogeography-based optimization (BBO) [14–16], differential evolution (DE) [17–19], particle swarm optimization (PSO) [20, 21], electromagnetism-like mechanism (EM) [22], and the charged system search algorithm (CSS) [23–25].

In 2010, Kaveh and Talatahari have firstly proposed a robust metaheuristic search technique, namely, CSS algorithm [23], for possibly nonlinear functions. The governing laws from the physics initiate the base of the CSS algorithm. CSS is a multiagent algorithm in which each agent is

considered as a charged sphere. Since these agents are treated as charged particles that can affect each other according to the Coulomb and Gauss laws from electrostatics, they are called charged particles (CPs). After determining the resultant force affected on each CP, the Newtonian motion law is utilized to determine the movement of the agents. The successive moving of CPs considering the resultant forces directs the agents toward optimum solutions.

The contribution of this paper is to present a hybrid CSS-based algorithm to find a seismic optimum design of steel frames considering four performance levels. The nonlinear analysis is required to reach the structural response at various performance levels. Therefore, the refined plastic hinge analysis method is developed to estimate the nonlinear behavior of the entire structural system and members effectively.

The organization of this paper is as follows. Section 2 and Section 3 describe the statement of the problem and the utilized analyses method, respectively. Our proposed CSS-based hybrid method is described in detail in Sections 4 and 5. Subsequently, the merits of our method are verified by numerical examples in Section 6. At last, Section 7 summarizes our work.

## 2. Statement of Seismic Design of Frames

The mathematical formulation of the structural optimization problems can be expressed as minimizing the weight of structures as the cost function without taking into consideration other influencing tributary parameters:

$$\text{Minimize : } W(X) = \sum_{j=1}^{ne} \rho \cdot L_j \cdot A_j, \quad (1)$$

where  $W(X)$  is the weight of the structure;  $X$  is the vector of design variables taken from W-shaped sections found in the AISC design manual [26];  $ne$  is the number of members;  $\rho$  is the material mass density; and  $L_j$  and  $A_j$  are the length and the cross-sectional area of the member  $j$ , respectively.

Lateral deflections of a building may cause human discomfort and minor damage of nonstructural components. Extreme inelastic lateral deflections due to a severe earthquake can cause the failure of mechanical, electrical and plumbing systems or suspended ceilings and equipment to fall, thereby posing threats to the human life. This matter is considered as the constraint functions in this paper as [27]

$$\begin{aligned} \text{OP Level } \Delta^{\text{OP}}(X) &\leq \bar{\Delta}^{\text{OP}}, \\ \text{IO Level } \Delta^{\text{IO}}(X) &\leq \bar{\Delta}^{\text{IO}}, \\ \text{LS Level } \Delta^{\text{LS}}(X) &\leq \bar{\Delta}^{\text{LS}}, \\ \text{CP Level } \Delta^{\text{CP}}(X) &\leq \bar{\Delta}^{\text{CP}}, \end{aligned} \quad (2)$$

where  $\Delta^{\text{level}}$  is the lateral drift and  $\bar{\Delta}^{\text{level}}$  is the allowable lateral drift (0.4%, 0.7%, 2.5% and 5% of the height of the building are taken as the allowable roof drifts for the OP, IO, LS, and CP performance levels, resp.). Here, OP, IO, LS, and CP are the different performance levels. Operational (OP), Immediate Occupancy (IO), life safety (LS), and collapse Prevention (CP) (FEMA-273, 1997), [28] are building performance levels. The operational level is that at which a building has sustained minimal or no damage to its structural and nonstructural components, and the building is suitable for normal occupancy or use; a building at the immediate occupancy level has sustained minimal or no damage to its structural elements and only minor damage to its nonstructural components, and is safe to be reoccupied immediately; a building at the life safety level has experienced extensive damage to its structural and nonstructural components and, while the risk to life is low, repairs may be required before reoccupancy can occur; the collapse prevention level is when a building has reached a state of impending partial or total collapse, where the building may have suffered a significant loss of lateral strength and stiffness with some permanent lateral deformation, but the major components of the gravity load carrying system should still continue to carry gravity load demands.

## 3. Pushover Analysis for Performance-Based Design

There are various methods of static pushover analyses to predict the seismic demands on building frameworks under equivalent static earthquake loading [29–35]; however, here a developed computer-based pushover analysis procedure is utilized [27] which was originally conceived for the elastic analysis of steel frameworks with semirigid connections [36, 37]. The analysis process is inspired from second-order inelastic analysis of semi-rigid framed structures that rigidity factor is replaced with plasticity factor in stiffness matrix. Fictitious plastic-hinge connections are necessary at the two ends of beam-column elements and semi-rigid analysis techniques were modified for the nonlinear load-deformation analysis of building frameworks under increasing seismic loads. The value of plasticity factor  $p$  is conceived from rigidity factor used in semi-rigid analysis. This factor  $r_i$  defines the rotational stiffness of the connection and can be interpreted as the ratio of the end-rotation  $\alpha_i$  of the member to the combined rotation  $\theta_i$  of the member as

$$r_i = \frac{\alpha_i}{\theta_i} = \frac{1}{1 + (3EI/RL)} \quad (i = 1, 2), \quad (3)$$

where  $R$  is the rotational stiffness of connection  $i$  and  $EI$  and  $L$  are the bending stiffness and length of the connected member, respectively. In fact, upon replacing connection rotational stiffness  $R$  with section postelastic flexural stiffness in (3), the degradation of the flexural stiffness of a member section experiencing postelastic behavior can be characterized by the plasticity factor:

$$p = \frac{1}{1 + (3EI/R^pL)}, \quad (4)$$

where  $R^p = dM/d\phi$  is the section postelastic flexural stiffness and  $p$  is the plasticity factor:

$$\% \text{plasticity} = 100(1 - p). \quad (5)$$

Here, the elastic stiffness matrix is comprised of both the first-order and the second-order geometric properties:

$$K = S_e C_e + S_g C_g. \quad (6)$$

The matrix  $K$  consists of two parts; the first part is conceived from Monfortoon and Wu's method [38] that employs the rigidity factor concept to develop a first-order elastic analysis technique for semi-rigid frames (i.e.,  $S_e \times C_e$ ) and the second part is conceived from Xu's method [36] that considers the rigidity-factor concept to develop a second-order elastic analysis technique for semi-rigid frames (i.e.,  $S_g \times C_g$ ). Here  $S_e$  and  $S_g$  are the standard first-order elastic and the second-order geometric stiffness matrices, respectively, when the member has rigid moment-connections;  $C_e$  and  $C_g$  are the corresponding correction matrices which account for the reduced rotational stiffness of the semi-rigid moment-connections. The flowchart of pushover analysis for performance-based design is shown in Figure 1.

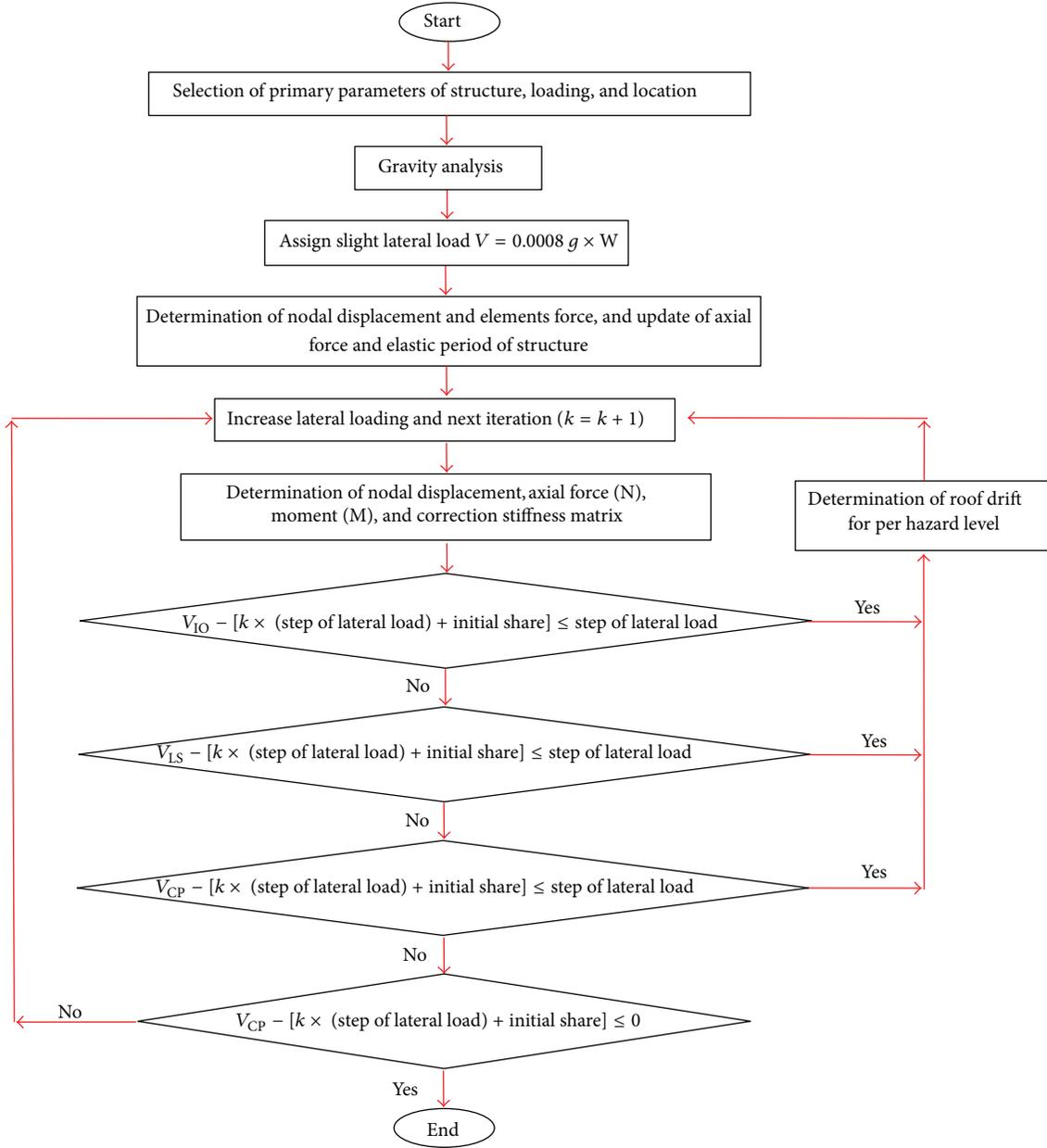


FIGURE 1: Flowchart of pushover analysis for performance-based design.

#### 4. Utilized Algorithms

A review of utilized metaheuristic algorithms is presented in the following subsections.

**4.1. Charged System Search Algorithm.** The charged system search (CSS) algorithm is based on the Coulomb and Gauss laws from electrical physics and the governing laws of motion from the Newtonian mechanics. This algorithm can be considered as a multiagent approach, where each agent is a charged particle (CP). Each CP is considered as a charged

sphere with radius  $a$ , having a uniform volume charge density, and is equal to [23]

$$q_j = \frac{W_j - W_{\text{worst}}}{W_{\text{best}} - W_{\text{worst}}}, \quad j = 1, 2, \dots, N, \quad (7)$$

where  $W_{\text{best}}$  and  $W_{\text{worst}}$  are the minimum and the maximum weight among all the particles,  $W_j$  represents the weight of the agent  $i$ , and  $N$  is the total number of CPs.

CPs can impose electrical forces on the others. The kind of the forces is attractive and its magnitude for the CP

located in the inside of the sphere is proportional to the separation distance between the CPs and for a CP located outside the sphere is inversely proportional to the square of the separation distance between the particles:

$$\mathbf{F}_j = q_j \sum_{i, i \neq j} \left( \frac{q_i}{a^3} r_{ij} \cdot \mathbf{i}_1 + \frac{q_i}{r_{ij}^2} \cdot \mathbf{i}_2 \right) P_{ij} (\mathbf{X}_i - \mathbf{X}_j),$$

$$j = 1, 2, \dots, N, \quad (8)$$

$$\mathbf{i}_1 = 1, \quad \mathbf{i}_2 = 0 \iff r_{ij} < a,$$

$$\mathbf{i}_1 = 0, \quad \mathbf{i}_2 = 1 \iff r_{ij} \geq a,$$

where  $\mathbf{F}_j$  is the resultant force acting on the  $j$ th CP and  $r_{ij}$  is the separation distance between two charged particles which is defined as follows:

$$r_{ij} = \frac{\|\mathbf{X}_i - \mathbf{X}_j\|}{\|(\mathbf{X}_i + \mathbf{X}_j)/2 - \mathbf{X}_{\text{best}}\| + \varepsilon}, \quad (9)$$

where  $\mathbf{X}_i$  and  $\mathbf{X}_j$  are the positions of the  $i$ th and  $j$ th CPs, respectively;  $\mathbf{X}_{\text{best}}$  is the position of the best current CP with the minimal weight; and  $\varepsilon$  is a small positive number. The initial positions of CPs are determined randomly in the search space and the initial velocities of charged particles are assumed to be zero.  $P_{ij}$  determines the probability of moving each CP toward the others as

$$P_{ij} = \begin{cases} 1 & \frac{W_i - W_{\text{best}}}{W_j - W_i} > \text{rand or } W_j > W_i \\ 0 & \text{otherwise.} \end{cases} \quad (10)$$

The resultant forces and the motion laws determine the new location of the CPs. At this stage, each CP moves toward to its new position considering the resultant forces and its previous velocity as

$$\mathbf{X}_{j,\text{new}} = \text{rand}_{j1} \cdot k_a \cdot \frac{\mathbf{F}_j}{m_j} \cdot \Delta t^2$$

$$+ \text{rand}_{j2} \cdot k_v \cdot \mathbf{V}_{j,\text{old}} \cdot \Delta t + \mathbf{X}_{j,\text{old}}, \quad (11)$$

$$\mathbf{V}_{j,\text{new}} = \frac{\mathbf{X}_{j,\text{new}} - \mathbf{X}_{j,\text{old}}}{\Delta t},$$

where  $k_a$  is the acceleration coefficient;  $k_v$  is the velocity coefficient to control the influence of the previous velocity; and  $\text{rand}_{j1}$  and  $\text{rand}_{j2}$  are two random numbers uniformly distributed in the range of (0, 1). If each CP exits from the allowable search space, its position is corrected using the harmony search-based handling approach as described by Kaveh and Talatahari [39]. In addition, to save the best design, a memory (charged memory) is considered containing the CMS number of positions for the so far best agents.

Both CSS and EM [22] are based on the governing laws from the electrical physics; however, the movement strategies, the resultant force for each agent, and deification of electrical charges for agents are different. The CSS algorithm utilizes

a velocity term while in the EM we have no term of a velocity. The EM just uses the Coulomb law to determine the forces while the CSS approach uses the Coulomb law as well as Gauss's law to explore the search space more efficiently. After evaluating the total force vector in the EM, each agent is moved in the direction of the force by a random step length (being uniformly distributed between 0 and 1) while the movements in the CSS are based on the governing laws of motion from the Newtonian mechanics. The potency of the EM is summarized to find the direction of an agent's movement, while in the CSS not only the directions but also the amount of movements are determined.

From the above discussion, it can be concluded that the CSS algorithm is a general form of the EM which contains its superiorities and avoids its disadvantages.

**4.2. Particle Swarm Optimization.** The particle swarm optimization (PSO) is motivated from the social behavior of bird flocking and fish schooling which has a population of individuals, called particles, that adjust their movements depending on both their own experience and the population's experience [20]. In other words, each particle in the PSO algorithm continuously focuses and refocuses on the effort of its search according to both local best and global best. In PSO, the position of each agent,  $\mathbf{X}_i^k$ , and its velocity,  $\mathbf{V}_i^{k+1}$ , are calculated as

$$\mathbf{X}_i^{k+1} = \mathbf{X}_i^k + \mathbf{V}_i^{k+1}, \quad (12)$$

$$\mathbf{V}_i^{k+1} = \omega \mathbf{V}_i^k + c_1 r_1 \circ (\mathbf{P}_i^k - \mathbf{X}_i^k) + c_2 r_2 \circ (\mathbf{P}_g^k - \mathbf{X}_i^k),$$

where  $\omega$  is an inertia weight to control the influence of the previous velocity,  $r_1$  and  $r_2$  are two random vectors uniformly distributed in the range of (0, 1), and  $c_1$  and  $c_2$  are two acceleration constants, and the sign "o" denotes element-by-element multiplication. The abovementioned formulations of the PSO algorithm can be combined and rewritten as

$$\mathbf{X}_i^{k+1} = \mathbf{X}_i^k + \omega \mathbf{V}_i^k + c_1 r_1 \circ (\mathbf{P}_i^k - \mathbf{X}_i^k) + c_2 r_2 \circ (\mathbf{P}_g^k - \mathbf{X}_i^k). \quad (13)$$

In some previous studies, to improve the performance of the algorithm, another term is added to the above formulae as

$$\mathbf{X}_i^{k+1} = \mathbf{X}_i^k + \omega \mathbf{V}_i^k + c_1 r_1 \circ (\mathbf{P}_i^k - \mathbf{X}_i^k)$$

$$+ c_2 r_2 \circ (\mathbf{P}_g^k - \mathbf{X}_i^k) + \sum_{j=1}^{ne} c_j r_j \circ (\mathbf{R}_j^k - \mathbf{X}_i^k), \quad (14)$$

where  $c_j$ , similar to  $c_1$  and  $c_2$ , is a constant value and  $r_j$  is a random vector.  $ne$  denotes the number of extra terms considered in the algorithm and  $\mathbf{R}_j^k$  is defined based on the type of the algorithm being used.

## 5. A Hybrid Optimization Algorithm

In the present hybrid algorithm, the advantage of the PSO containing utilizing the local best and the global best is added

to the CSS algorithm. The charged memory (CM) for the hybrid algorithm is treated as the local best in the PSO, and the CM updating process is defined as

$$CM_{i,new} = \begin{cases} CM_{i,old}, & W(\mathbf{X}_{i,new}) \geq W(CM_{i,old}), \\ \mathbf{X}_{i,new}, & W(\mathbf{X}_{i,new}) < W(CM_{i,old}), \end{cases} \quad (15)$$

in which the first term identifies that when the new position is not better than the previous one the updating does not perform while when the new position is better than the stored so far good position the new solution vector is replaced. In the first iteration, the vector stored in CM and the first positions of the agents will be identical. Considering the abovementioned new charged memory, the electric forces generated by agents are modified as

$$\begin{aligned} \mathbf{F}_j = & \sum_{i \in S_1} \left( \frac{q_i}{a^3} r_{ij} \cdot i_1 + \frac{q_i}{r_{ij}^2} \cdot i_2 \right) (CM_{i,old} - \mathbf{X}_j) \\ & + \sum_{i \in S_2} \left( \frac{q_i}{a^3} r_{ij} \cdot i_1 + \frac{q_i}{r_{ij}^2} \cdot i_2 \right) ar_{ij} p_{ij} (\mathbf{X}_i - \mathbf{X}_j), \end{aligned} \quad (16)$$

where  $S_1$  and  $S_2$  are defined as follows:

$$S_1 = \{t_1, t_2, \dots, t_n \mid q(t) > q(j), j = 1, 2, \dots, N, j \neq i, g\}, \quad (17)$$

$$S_2 = S - S_1, \quad (18)$$

in which  $S_1$  determines the set of agents utilized from CM,  $n$  denotes the number of CM agents,  $S$  is utilized as a set of all agents' number, and thus  $S_2$  will be the set of current agents used for directing the agent  $j$ . Here, in the primary iterations  $n$  is set to two continuing the number of the best stored so far agent among all CPs (global best) and  $j$ th agent stored in the CM which is treated as local best. Then the number of used agents from CM is increased linearly and finally it reached  $N$  in the last iterations. In this hybrid algorithm,  $CM_{i,old}$  will be treated similar to  $\mathbf{P}_i^k$  in the PSO. The other modification is that the forces can be attractive or repulsive, and  $ar_{ij}$  is added to fulfill this aim which determines the kind of the force as

$$ar_{ij} = \begin{cases} +1, & \text{w.p. } k_t, \\ -1, & \text{w.p. } 1 - k_t, \end{cases} \quad (19)$$

where "w.p." represents the abbreviation for "with the probability" and  $k_t$  is a parameter to control the effect of the kind of forces. Comparing to (10), this new formula (18) considers the best so far location of agents and the best local position of the current agent in addition to the location of other agents. Also, here  $m_j$  is assumed to be  $q_j$  and therefore (12) is simplified as

$$\mathbf{X}_{j,new} = k_a \cdot r_1 \cdot \mathbf{F}_j + k_v \cdot r_2 \cdot \mathbf{V}_{j,old} + \mathbf{X}_{j,old}. \quad (20)$$

The pseudocode of the hybrid algorithm can be summarized as follows.

*Step 1* (initialization). The magnitude of the charge for each CP is defined by (7). The initial positions of the CPs are determined randomly and the initial velocities of charged particles are assumed to be zero.

*Step 2* (CM creation). The position of the initial agents and the values of their corresponding objective functions are saved in the charged memory (CM).

*Step 3* (the forces determination). The probability of moving each CP towards the others ( $p_{ij}$ ) and the kind of forces ( $a_{ij}$ ) are determined using (10) and (19), respectively, and the resultant force vector for each CP is calculated using (18).

*Step 4* (solution construction). Each CP moves to the new position according to (20).

*Step 5* (CM updating). CM updating is performed according to (15).

*Step 6* (terminating criterion control). Steps 3–5 are repeated for a predefined number of iterations.

## 6. Design Examples

Two building frameworks are selected for seismic optimum design using the metaheuristic algorithm [27]. These frames have previously been used to illustrate the pushover analysis technique by Hasan et al. [42] and Talatahari [40].

The expected yield strength of steel material used for column members is  $\sigma_{ye} = 397$  MPa, while  $\sigma_{ye} = 339$  MPa is considered for beam members. The constant gravity load  $w$  is accounted for a tributary-area width of 4.57 m and dead-load and live-load factors of 1.2 and 1.6, respectively. For each example, 30 independent runs are carried out using the new hybrid algorithms and compared with other algorithms. The number of 20 individuals for CPs is used and the values of constants  $k_v$  and  $k_a$  are set to 0.4.

*6.1. Four-Bay Three-Story Steel Frame.* The configuration, grouping of the members and applied loads of the four-bay three-story framed structure are shown in Figure 2, [27]. The 27 members, of the structure are categorized into five groups, as indicated in the figure. The modulus of elasticity is taken as  $E = 200$  GPa. The constant gravity load of  $w_1 = 32$  kN/m is applied to the first and second story beams, while the gravity load of  $w_2 = 28.7$  kN/m is applied to the roof beams. The seismic weight is 4,688 kN for each of the first and second stories and 5,071 kN for the roof story.

The performance-based optimum results for the metaheuristic algorithm are summarized in Table 1. The hybrid CSS, HPACO, ACO, and GA need 4500, 4500, 3900 and 6800 analyses to reach a convergence while 8500 analyses required by the PSO. The best hybrid CSS design results in a frame that weighs 273.7 kN, which is lighter than the design of Gall optimization algorithm. The result of conventional design [41] is approximately 50% more than the result of new algorithm. In a series of 30 different design runs, the average weight of the hybrid CSS designs is 286.7 kN, with a standard deviation of 5.651 kN, while the average weight of the PSACO, PSO, and ACO designs is 290.4 kN, 302.4 kN, and 294.3 kN, respectively. The standard deviation values are 6.45 kN, 10.45 kN and 7.56 for the PSACO, PSO, and ACO, respectively.

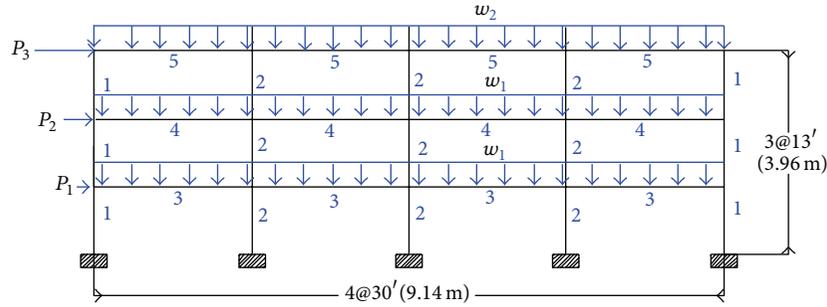


FIGURE 2: Three-story steel moment-frame.

TABLE 1: The statistical information of performance-based optimum designs for the 4-bay 3-story frame.

Algorithm	Hybrid CSS	PSACO [40]	PSO [40]	ACO [27]	GA [27]	A conventional design [41]
Best weight (kN)	273.7	279.2	286.3	283.4	303.9	412.9 kN
Average weight (kN)	286.7	290.4	302.4	294.3	321.5	—
Worst weight (kN)	297.8	298.5	310.7	303.2	339.7	—
Std. dev. (kN)	5.651	6.453	10.453	7.566	14.332	—
Average number of analyses	4,500	4,500	8,500	3,900	6,800	—

TABLE 2: The statistical information of performance-based optimum designs for the 4-bay 9-story frame.

Algorithm	Hybrid CSS	PSACO [40]	PSO [40]	ACO [27]	GA [27]
Best weight (kN)	1568.66	1601.32	1682.63	1631.83	1723.1
Average weight (kN)	1626.32	1650.55	1725.36	1696.2	1791.4
Worst weight (kN)	1725.36	1759.65	1813.25	1786.94	1943.2
Std. dev. (kN)	30.35	38.52	66.35	49.33	78.33
Average number of analyses	5,000	6,000	12,500	5,600	9,700

6.2. *Five-Bay Nine-Story Steel Frame.* A five-bay nine-story steel frame is considered as shown in Figure 3. The material has a modulus of elasticity equal to  $E = 200$  GPa. The 108 members of the structure are categorized into fifteen groups, as indicated in the figure. The constant gravity load of  $w_1 = 32$  kN/m is applied to the beams in the first to the eighth story, while  $w_2 = 28.7$  kN/m is applied to the roof beams. The seismic weights are 4,942 kN for the first story, 4,857 kN for each of the second to eighth stories, and 5,231 kN for the roof story. In this example, each of the five beam element groups is chosen from all 267 W-shapes, while the eight column element groups are limited to W14 sections (37 W-shapes).

Table 2 presents the statistical results obtained by the metaheuristic algorithms. The best hybrid CSS design results in a frame weighing 1568.66 kN which is 1.9%, 7.0%, 3.8%, and 9.5% lighter than the PSACO, PSO, ACO, and GA. In order to converge to a solution for the hybrid CSS algorithm, approximately 5,000 frame analyses are required which are less than the 6,000, 12,500, and 9,700 analyses necessary for the PSACO, PSO, and GA, respectively. The ACO needs only 5,600 analyses to find an optimum result.

## 7. Conclusion Remarks

The problem of optimum design of frame structures is formulated to minimize the weight of the structure considering

the required constraints specified by design codes. For seismic design of structures, two main points should be considered: structural costs and structural damages. As a result, it is essential to control the lateral drift of building frameworks under seismic loading at various performance levels. To fulfill this aim, in this paper a hybrid optimization method is presented. The algorithm is based on the CSS algorithm. CSS is a multiagent algorithm in which each agent is considered as a charged sphere. Since these agents are treated as charged particles that can affect each other according to the Coulomb and Gauss laws from electrostatics, in the present hybrid algorithm, the advantage of the PSO containing utilizing the local best and the global best is added to the CSS algorithm. The charged memory for the hybrid algorithm is treated as the local best in the PSO, and the CM updating process is redefined to adapt the new requirements.

A simple computer-based method for push-over analysis of steel building frameworks subject to equivalent-static earthquake loading is utilized. The method accounts for first-order elastic and second-order geometric stiffness properties and the influence that combined stresses have on plastic behavior and employs a conventional elastic analysis procedure modified by a plasticity-factor to trace elastic-plastic behavior over the range of performance levels for a structure [27]. Two examples are optimized using the new algorithm as

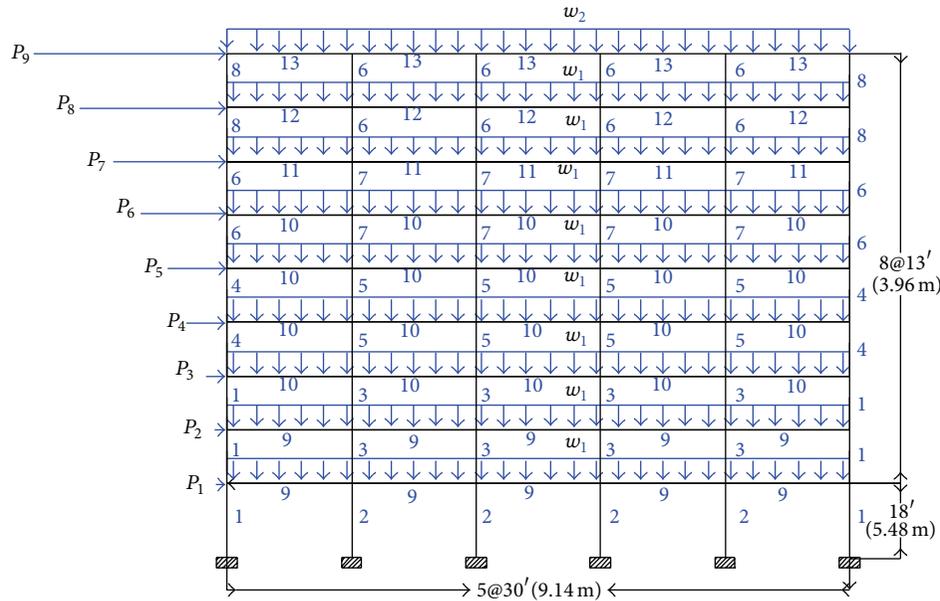


FIGURE 3: Nine-story steel moment-frame.

well as some advanced metaheuristic algorithms to investigate the capability of the new method. The genetic algorithm, ant colony optimization, particle swarm optimization, and particle swarm ant colony optimization method as well as the new hybrid method are utilized to find optimum seismic design of examples. The obtained results indicate that the new algorithm compared to GA, ACO, PSO, and PSACO can find better optimum seismic design of structures.

### Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

### References

- [1] M. Dorigo and T. Stutzle, *Ant Colony Optimization*, MIT Press, Cambridge, Mass, USA, 2004.
- [2] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm," *Journal of Global Optimization*, vol. 39, no. 3, pp. 459–471, 2007.
- [3] X. Li and M. Yin, "Parameter estimation for chaotic systems by improved artificial bee colony algorithm," *Journal of Computational and Theoretical Nanoscience*, vol. 10, no. 3, pp. 756–762, 2013.
- [4] A. H. Gandomi, X.-S. Yang, and A. H. Alavi, "Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems," *Engineering with Computers*, vol. 29, no. 1, pp. 17–35, 2013.
- [5] G. Wang, L. Guo, H. Duan, H. Wang, L. Liu, and M. Shao, "A hybrid meta-heuristic DE/CS algorithm for UCAV three-dimension path planning," *The Scientific World Journal*, vol. 2012, Article ID 583973, 11 pages, 2012.

- [6] X. T. Li and M. H. Yin, "Parameter estimation for chaotic systems using the cuckoo search algorithm with an orthogonal learning method," *Chinese Physics B*, vol. 21, no. 5, Article ID 050507, 2012.
- [7] X. S. Yang and A. H. Gandomi, "Bat algorithm: a novel approach for global engineering optimization," *Engineering Computations*, vol. 29, no. 5, pp. 464–483, 2012.
- [8] G. Wang, L. Guo, H. Duan, L. Liu, and H. Wang, "Path planning for UCAV using bat algorithm with mutation," *The Scientific World Journal*, vol. 2012, Article ID 418946, 15 pages, 2012.
- [9] A. H. Gandomi and A. H. Alavi, "Multi-stage genetic programming: a new strategy to nonlinear system modeling," *Information Sciences*, vol. 181, no. 23, pp. 5227–5239, 2011.
- [10] H.-G. Beyer, *The theory of evolution strategies*, Natural Computing Series, Springer, Berlin, Germany, 2001.
- [11] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, New York, NY, USA, 1998.
- [12] Z. W. Geem, J. H. Kim, and G. V. Loganathan, "A new heuristic optimization algorithm: harmony search," *Simulation*, vol. 76, no. 2, pp. 60–68, 2001.
- [13] G. Wang and L. Guo, "A novel hybrid bat algorithm with harmony search for global numerical optimization," *Journal of Applied Mathematics*, vol. 2013, Article ID 696491, 21 pages, 2013.
- [14] D. Simon, "Biogeography-based optimization," *IEEE Transactions on Evolutionary Computation*, vol. 12, no. 6, pp. 702–713, 2008.
- [15] H. Duan, W. Zhao, G. Wang, and X. Feng, "Test-sheet composition using analytic hierarchy process and hybrid metaheuristic algorithm TS/BBO," *Mathematical Problems in Engineering*, vol. 2012, Article ID 712752, 22 pages, 2012.
- [16] X. Li, J. Wang, J. Zhou, and M. Yin, "A perturb biogeography based optimization with mutation for global numerical optimization," *Applied Mathematics and Computation*, vol. 218, no. 2, pp. 598–609, 2011.

- [17] R. Storn and K. Price, "Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization. An International Journal Dealing with Theoretical and Computational Aspects of Seeking Global Optima and Their Applications in Science, Management and Engineering*, vol. 11, no. 4, pp. 341–359, 1997.
- [18] A. H. Gandomi, X.-S. Yang, S. Talatahari, and S. Deb, "Coupled eagle strategy and differential evolution for unconstrained and constrained global optimization," *Computers & Mathematics with Applications*, vol. 63, no. 1, pp. 191–200, 2012.
- [19] X. Li and M. Yin, "Self-adaptive constrained artificial bee colony for constrained numerical optimization," *Neural Computing and Applications*, 2012.
- [20] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, pp. 1942–1948, Perth, Australia, December 1995.
- [21] S. Talatahari, M. Kheirollahi, C. Farahmandpour, and A. H. Gandomi, "A multi-stage particle swarm for optimum design of truss structures," *Neural Computing & Applications*, vol. 23, no. 5, pp. 1297–1309, 2013.
- [22] S. Birbil and S.-C. Fang, "An electromagnetism-like mechanism for global optimization," *Journal of Global Optimization*, vol. 25, no. 3, pp. 263–282, 2003.
- [23] A. Kaveh and S. Talatahari, "A novel heuristic optimization method: charged system search," *Acta Mechanica*, vol. 213, no. 3, pp. 267–289, 2010.
- [24] S. Talatahari, A. Kaveh, and N. Mohajer Rahbari, "Parameter identification of Bouc-Wen model for MR fluid dampers using adaptive charged system search optimization," *Journal of Mechanical Science and Technology*, vol. 26, no. 8, pp. 2523–2534, 2012.
- [25] A. Kaveh and S. Talatahari, "An enhanced charged system search for configuration optimization using the concept of fields of forces," *Structural and Multidisciplinary Optimization*, vol. 43, no. 3, pp. 339–351, 2011.
- [26] Manual of steel construction, *Load and Resistance Factor Design*, American Institute of Steel Construction, Chicago, Ill, USA, 2001.
- [27] A. Kaveh, B. Farahmand Azar, A. Hadidi, F. Rezazadeh Sorochi, and S. Talatahari, "Performance-based seismic design of steel frames using ant colony optimization," *Journal of Constructional Steel Research*, vol. 66, no. 4, pp. 566–574, 2010.
- [28] Federal Emergency Management Agency, *FEMA-273. NEHRP Guideline for the Seismic Rehabilitation of Buildings*, Building Seismic Safety Council, Washington, DC, USA, 1997.
- [29] R. S. Lawson, V. Vance, and H. Krawinkler, "Nonlinear static push-over analysis why, when, and how?" in *Proceedings of 5th U.S. National Conference on Earthquake Engineering*, vol. 1, pp. 283–292, EERI, Chicago, Ill, USA, 1994.
- [30] A. C. Biddah and N. Naumoski, "Use of pushover test to evaluate damage of reinforced concrete frame structures subjected to strong seismic ground motions," in *Proceedings of 7th Canadian Conference on Earthquake Engineering*, Montreal, Canada, 1995.
- [31] A. S. Moghadam and W. K. Tso, "3-D pushover analysis for eccentric buildings," in *Proceedings of 7th Canadian Conference on Earthquake Engineering*, Montreal, Canada, 1995.
- [32] A. Ferhi and K. Z. Truman, "Behaviour of asymmetric building systems under a monotonic load—I," *Engineering Structures*, vol. 18, no. 2, pp. 133–141, 1996.
- [33] A. Ferhi and K. Z. Truman, "Behaviour of asymmetric building systems under a monotonic load—II," *Engineering Structures*, vol. 18, no. 2, pp. 142–153, 1996.
- [34] J. M. Bracci, S. K. Kunnath, and A. M. Reinhorn, "Seismic performance and retrofit evaluation of reinforced concrete structures," *Journal of Structural Engineering*, vol. 123, no. 1, pp. 3–10, 1997.
- [35] V. Kilar and P. Fajfar, "Simple push-over analysis of asymmetric buildings," *Earthquake Engineering and Structural Dynamics*, vol. 26, no. 2, pp. 233–249, 1997.
- [36] L. Xu, "Geometrical stiffness and sensitivity matrices for optimization of semi-rigid steel frameworks," *Structural Optimization*, vol. 5, no. 1-2, pp. 95–99, 1992.
- [37] L. Xu and D. E. Grierson, "Computer-automated design of semirigid steel frameworks," *Journal of Structural Engineering New York, N.Y.*, vol. 119, no. 6, pp. 1740–1760, 1993.
- [38] G. R. Monfortoon and T. S. Wu, "Matrix analysis of semi-rigidly connected steel frames," *Journal of Structural Engineering*, vol. 89, no. 6, pp. 13–42, 1963.
- [39] A. Kaveh and S. Talatahari, "Particle swarm optimizer, ant colony strategy and harmony search scheme hybridized for optimization of truss structures," *Computers and Structures*, vol. 87, no. 5-6, pp. 267–283, 2009.
- [40] S. Talatahari, "Optimum performance-based seismic design of frames using metaheuristic optimization algorithms," in *Metaheuristics in Water, Geotechnical and Transport Engineering*, X.-S. Yang, A. H. Gandomi, S. Talatahari, and A. H. Alavi, Eds., Elsevier, 2012.
- [41] R. Hassan, B. Cohaniam, O. De Weck, and G. Venter, "A comparison of particle swarm optimization and the genetic algorithm," in *Proceedings of the 46th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, pp. 18–21, April 2005.
- [42] R. Hasan, L. Xu, and D. E. Grierson, "Push-over analysis for performance-based seismic design," *Computers and Structures*, vol. 80, no. 31, pp. 2483–2493, 2002.

## Research Article

# Incremental Tensor Principal Component Analysis for Handwritten Digit Recognition

**Chang Liu,<sup>1,2</sup> Tao Yan,<sup>1,2</sup> WeiDong Zhao,<sup>1,2</sup> YongHong Liu,<sup>1,2</sup> Dan Li,<sup>1,2</sup>  
Feng Lin,<sup>3</sup> and JiLiu Zhou<sup>3</sup>**

<sup>1</sup> College of Information Science and Technology, Chengdu University, Chengdu 610106, China

<sup>2</sup> Key Laboratory of Pattern Recognition and Intelligent Information Processing, Institutions of Higher Education of Sichuan Province, Chengdu 610106, China

<sup>3</sup> School of Computer Science, Sichuan University, Chengdu 610065, China

Correspondence should be addressed to YongHong Liu; 284424241@qq.com

Received 5 July 2013; Revised 21 September 2013; Accepted 22 September 2013; Published 30 January 2014

Academic Editor: Praveen Agarwal

Copyright © 2014 C. Liu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

To overcome the shortcomings of traditional dimensionality reduction algorithms, incremental tensor principal component analysis (ITPCA) based on updated-SVD technique algorithm is proposed in this paper. This paper proves the relationship between PCA, 2DPCA, MPCA, and the graph embedding framework theoretically and derives the incremental learning procedure to add single sample and multiple samples in detail. The experiments on handwritten digit recognition have demonstrated that ITPCA has achieved better recognition performance than that of vector-based principal component analysis (PCA), incremental principal component analysis (IPCA), and multilinear principal component analysis (MPCA) algorithms. At the same time, ITPCA also has lower time and space complexity.

## 1. Introduction

Pattern recognition and computer vision require processing a large amount of multi-dimensional data, such as image and video data. Until now, a large number of dimensionality reduction algorithms have been investigated. These algorithms project the whole data into a low-dimensional space and construct new features by analyzing the statistical relationship hidden in the data set. The new features often give good information or hints about the data's intrinsic structure. As a classical dimensionality reduction algorithm, principal component analysis has been applied in various applications widely.

Traditional dimensionality reduction algorithms generally transform each multi-dimensional data into a vector by concatenating rows, which is called Vectorization. Such kind of the vectorization operation has largely increased the computational cost of data analysis and seriously destroys the intrinsic tensor structure of high-order data. Consequently, tensor dimensionality reduction algorithms are developed

based on tensor algebra [1–10]. Reference [10] has summarized existing multilinear subspace learning algorithms for tensor data. Reference [11] has generalized principal component analysis into tensor space and presented multilinear principal component analysis (MPCA). Reference [12] has proposed the graph embedding framework to unify all dimensionality reduction algorithms.

Furthermore, traditional dimensionality reduction algorithms generally employ off-line learning to deal with new added samples, which aggravates the computational cost. To address this problem, on-line learning algorithms are proposed [13, 14]. In particular, reference [15] has developed incremental principal component analysis (IPCA) based on updated-SVD technique. But most on-line learning algorithms focus on vector-based methods, only a limited number of works study incremental learning in tensor space [16–18].

To improve the incremental learning in tensor space, this paper presents incremental tensor principal component analysis (ITPCA) based on updated-SVD technique combining tensor representation with incremental learning.

This paper proves the relationship between PCA, 2DPCA, MPCA, and the graph embedding framework theoretically and derives the incremental learning procedure to add single sample and multiple samples in detail. The experiments on handwritten digit recognition have demonstrated that ITPCA has achieved better performance than vector-based incremental principal component analysis (IPCA) and multilinear principal component analysis (MPCA) algorithms. At the same time, ITPCA also has lower time and space complexity than MPCA.

## 2. Tensor Principal Component Analysis

In this section, we will employ tensor representation to express high-dimensional image data. Consequently, a high-dimensional image dataset can be expressed as a tensor dataset  $X = \{X_1, \dots, X_M\}$ , where  $X_i \in \mathbb{R}^{I_1 \times \dots \times I_N}$  is an  $N$  dimensional tensor and  $M$  is the number of samples in the dataset. Based on the representation, the following definitions are introduced.

*Definition 1.* For tensor dataset  $X$ , the mean tensor is defined as follows:

$$\bar{X} = \frac{1}{M} \sum_{i=1}^M X_i \in \mathbb{R}^{I_1 \times \dots \times I_N}. \quad (1)$$

*Definition 2.* The unfolding matrix of the mean tensor along the  $n$ th dimension is called the mode- $n$  mean matrix and is defined as follows:

$$\bar{X}^{(n)} = \frac{1}{M} \sum_{i=1}^M X_i^{(n)} \in \mathbb{R}^{I_n \times \prod_{i \neq n} I_i}. \quad (2)$$

*Definition 3.* For tensor dataset  $X$ , the total scatter tensor is defined as follows:

$$\Psi_X = \sum_{m=1}^M \|X_m - \bar{X}\|^2, \quad (3)$$

where  $\|A\|$  is the norm of the tensor.

*Definition 4.* For tensor dataset  $X$ , the mode- $n$  total scatter matrix is defined as follows:

$$C^{(n)} = \sum_{i=1}^M \left( X_i^{(n)} - \bar{X}^{(n)} \right) \left( X_i^{(n)} - \bar{X}^{(n)} \right)^T, \quad (4)$$

where  $\bar{X}^{(n)}$  is the mode- $n$  mean matrix and  $X_i^{(n)}$  is the mode- $n$  unfolding matrix of tensor  $X_i$ .

Tensor PCA is introduced in [11, 19]. The target is to compute  $N$  orthogonal projective matrices  $\{U^{(n)} \in \mathbb{R}^{I_n \times P_n}, n = 1, \dots, N\}$  to maximize the total scatter tensor of the projected low-dimensional feature as follows:

$$\begin{aligned} f \{U^{(n)}, n = 1, \dots, N\} &= \arg \max_{U^{(n)}} \Psi_y \\ &= \arg \max_{U^{(n)}} \sum_{m=1}^M \|Y_m - \bar{Y}\|^2, \end{aligned} \quad (5)$$

where  $Y_m = X_m \times_1 U^{(1)T} \times_2 U^{(2)T} \times \dots \times_{n-1} U^{(n-1)T} \times_{n+1} U^{(n+1)T} \times \dots \times_N U^{(N)T}$ .

Since it is difficult to solve  $N$  orthogonal projective matrices simultaneously, an iterative procedure is employed to approximately compute these  $N$  orthogonal projective matrices. Generally, since it is assumed that the projective matrices  $\{U^{(1)}, \dots, U^{(n-1)}, U^{(n+1)}, \dots, U^{(N)}\}$  are known, we can solve the following optimized problem to obtain  $U^{(n)}$ :

$$\arg \max_{U^{(n)}} \sum_{m=1}^M \left( C_m^{(n)} C_m^{(n)T} \right), \quad (6)$$

where  $C_m^{(n)} = (X_m - \bar{X}) \times_1 U^{(1)T} \times_2 U^{(2)T} \times \dots \times_{n-1} U^{(n-1)T} \times_{n+1} U^{(n+1)T} \times \dots \times_N U^{(N)T}$  and  $C_m^{(n)}$  is the mode- $n$  unfolding matrix of tensor  $C_m$ .

According to the above analysis, it is easy to derive the following theorems.

**Theorem 5** (see [11]). *For the order of tensor data  $n = 1$ , that is, for the first-order tensor, the objective function of MPCA is equal to that of PCA.*

*Proof.* For the first-order tensor,  $X_m \in \mathbb{R}^{I \times 1}$  is a vector, then (6) is

$$\sum_{m=1}^M \left( C_m^{(n)} C_m^{(n)T} \right) = \sum_{m=1}^M \left( U^T (X_m - \bar{X}) (X_m - \bar{X})^T U \right). \quad (7)$$

So MPCA for first-order tensor is equal to vector-based PCA.  $\square$

**Theorem 6** (see [11]). *For the order of tensor data  $n = 2$ , that is, for the second-order tensor, the objective function of MPCA is equal to that of 2DsPCA.*

*Proof.* For the second-order tensor,  $X_m \in \mathbb{R}^{I_1 \times I_2}$  is a matrix; it is needed to solve two projective matrices  $U^{(1)}$  and  $U^{(2)}$ , then (5) becomes

$$\sum_{m=1}^M \|Y_m - \bar{Y}\|^2 = \sum_{m=1}^M \|U^{(1)T} (X_m - \bar{X}) U^{(2)}\|^2. \quad (8)$$

The above equation exactly is the objective function of B2DPCA (bidirectional 2DPCA) [20–22]. Letting  $U^{(2)} = I$ , the projective matrix  $U^{(1)}$  is solved. In this case, the objective function is

$$\sum_{m=1}^M \|Y_m - \bar{Y}\|^2 = \sum_{m=1}^M \|U^{(1)T} (X_m - \bar{X}) I\|^2. \quad (9)$$

Then the above equation is simplified into the objective function of row 2DPCA [23, 24]. Similarly, letting  $U^{(1)} = I$ , the projective matrix  $U^{(2)}$  is solved; the objective function is

$$\sum_{m=1}^M \|Y_m - \bar{Y}\|^2 = \sum_{m=1}^M \|I^T (X_m - \bar{X}) U^{(2)}\|^2. \quad (10)$$

Then the above equation is simplified into the objective function of column 2DPCA [23, 24].  $\square$

Although vector-based and 2DPCA can be respected as the special cases of MPCA, MPCA and 2DPCA employ different techniques to solve the projective matrices. 2DPCA carries out PCA to row data and column data, respectively, and MPCA employs an iterative solution to compute  $N$  projective matrices. If it is supposed that the projective matrices  $\{U^{(1)}, \dots, U^{(n-1)}, U^{(n+1)}, \dots, U^{(N)}\}$  are known, then  $U^{(n)}$  is solved. Equation (6) can be expressed as follows:

$$\begin{aligned} C^{(n)} &= \sum_{i=1}^M \left( \left( X_i^{(n)} - \bar{X}^{(n)} \right) \times_k U^{(k)T} \Big|_{\substack{k=1 \\ k \neq n}}^N \right) \\ &\quad \times \left( \left( X_i^{(n)} - \bar{X}^{(n)} \right) \times_k U^{(k)T} \Big|_{\substack{k=1 \\ k \neq n}}^N \right)^T \\ &= \sum_{i=1}^M \left( \left( X_i^{(n)} - \bar{X}^{(n)} \right) U^{(-n)} \right) \left( \left( X_i^{(n)} - \bar{X}^{(n)} \right) U^{(-n)} \right)^T \\ &= \sum_{i=1}^M \left( \left( X_i^{(n)} - \bar{X}^{(n)} \right) U^{(-n)} U^{(-n)T} \left( X_i^{(n)} - \bar{X}^{(n)} \right)^T \right), \end{aligned} \quad (11)$$

where  $U^{(-n)} = U^{(N)} \dots \otimes U^{(n+1)} \otimes U^{(n-1)} \dots \otimes U^{(1)}$ .

Because

$$\begin{aligned} U^{(-n)} U^{(-n)T} &= \left( U^{(N)} \dots \otimes U^{(n+1)} \otimes U^{(n-1)} \dots \otimes U^{(1)} \right) \\ &\quad \times \left( U^{(N)} \dots \otimes U^{(n+1)} \otimes U^{(n-1)} \dots \otimes U^{(1)} \right)^T. \end{aligned} \quad (12)$$

Based on the Kronecker product, we can get the following:

$$\begin{aligned} (A \otimes B)^T &= A^T \otimes B^T, \\ (A \otimes B)(C \otimes D) &= AC \otimes BD. \end{aligned} \quad (13)$$

So

$$\begin{aligned} U^{(-n)} U^{(-n)T} &= U^{(N)} U^{(N)T} \dots \otimes U^{(n+1)} U^{(n+1)T} \\ &\quad \otimes U^{(n-1)} U^{(n-1)T} \dots \otimes U^{(1)} U^{(1)T}. \end{aligned} \quad (14)$$

Since  $U^{(i)} \in \mathbb{R}^{I_i \times I_i}$  is an orthogonal matrix,  $U^{(i)} U^{(i)T} = I$ ,  $i = 1, \dots, N$ ,  $i \neq n$ , and  $U^{(-n)} U^{(-n)T} = I$ .

If the dimensions of projective matrices do not change in iterative procedure, then

$$C^{(n)} = \sum_{i=1}^K \left( X_i^{(n)} - \bar{X}^{(n)} \right) \left( X_i^{(n)} - \bar{X}^{(n)} \right)^T. \quad (15)$$

The above equation is equal to B2DPCA. Because MPCA updates projective matrices during iterative procedure, it has achieved better performance than 2DPCA.

**Theorem 7.** MPCA can be unified into the graph embedding framework [12].

*Proof.* Based on the basic knowledge of tensor algebra, we can get the following:

$$\sum_{m=1}^M \|Y_m - \bar{Y}\|^2 = \sum_{m=1}^M \|\text{vec}(Y_m) - \text{vec}(\bar{Y})\|^2. \quad (16)$$

Letting  $y_m = \text{vec}(Y_m)$ ,  $\mu = \text{vec}(\bar{Y})$ , we can get the following:

$$\begin{aligned} &\sum_{i=1}^K \|y_i - \mu\|^2 \\ &= \sum_{i=1}^K (y_i - \mu)(y_i - \mu)^T \\ &= \sum_{i=1}^K \left( y_i - \frac{1}{N} \sum_{j=1}^N y_j \right) \left( y_i - \frac{1}{N} \sum_{j=1}^N y_j \right)^T \\ &= \sum_{i=1}^K \left( y_i y_i^T - \frac{1}{N} y_i \left( \sum_{j=1}^K y_j \right)^T - \frac{1}{N} \left( \sum_{j=1}^K y_j \right) y_i^T \right. \\ &\quad \left. + \frac{1}{N^2} \left( \sum_{j=1}^K y_j \right) \left( \sum_{j=1}^K y_j \right)^T \right) \\ &= \sum_{i=1}^K y_i y_i^T - \frac{1}{N} \sum_{i=1}^K y_i \left( \sum_{j=1}^K y_j \right)^T \\ &\quad - \frac{1}{N} \sum_{i=1}^K \sum_{j=1}^K y_j y_i^T + \frac{1}{N} \left( \sum_{j=1}^K y_j \right) \left( \sum_{j=1}^K y_j \right)^T \\ &= \sum_{i=1}^K y_i y_i^T - \frac{1}{N} \sum_{i=1}^K \sum_{j=1}^K y_i y_j^T \\ &= \sum_{i=1}^K \left( \sum_{j=1}^K W_{ij} \right) y_i y_i^T - \sum_{i,j=1}^K W_{ij} y_i y_j^T \\ &= \frac{1}{2} \sum_{i,j=1}^K W_{ij} (y_i y_i^T + y_j y_j^T - y_i y_j^T - y_j y_i^T) \\ &= \frac{1}{2} \sum_{i,j=1}^K W_{ij} (y_i - y_j)(y_i - y_j)^T \\ &= \frac{1}{2} \sum_{i,j=1}^K W_{ij} \|y_i - y_j\|_F^2, \end{aligned} \quad (17)$$

where the similarity matrix  $W \in \mathbb{R}^{M \times M}$ ; for any  $i, j$ , we have  $W_{ij} = 1/K$ . So (16) can be written as follows:

$$\begin{aligned} & \sum_{m=1}^M \|Y_m - \bar{Y}\|^2 \\ &= \frac{1}{2} \sum_{i,j=1}^M W_{ij} \|Y_i - Y_j\|^2 \\ &= \frac{1}{2} \sum_{i,j=1}^M W_{ij} \left\| X_i \times_n U^{(n)} \Big|_{n=1}^N - X_j \times_n U^{(n)} \Big|_{n=1}^N \right\|^2. \end{aligned} \quad (18)$$

So the theorem is proved.  $\square$

### 3. Incremental Tensor Principal Component Analysis

**3.1. Incremental Learning Based on Single Sample.** Given initial training samples  $X_{\text{old}} = \{X_1, \dots, X_K\}$ ,  $X_i \in \mathbb{R}^{I_1 \times \dots \times I_N}$ , when a new sample  $X_{\text{new}} \in \mathbb{R}^{I_1 \times \dots \times I_N}$  is added, the training dataset becomes  $X = \{X_{\text{old}}, X_{\text{new}}\}$ .

The mean tensor of initial samples is

$$\bar{X}_{\text{old}} = \frac{1}{K} \sum_{i=1}^K X_i. \quad (19)$$

The covariance tensor of initial samples is

$$C_{\text{old}} = \sum_{i=1}^K \|X_i - \bar{X}_{\text{old}}\|^2. \quad (20)$$

The mode- $n$  covariance matrix of initial samples is

$$C_{\text{old}}^{(n)} = \sum_{i=1}^K (X_i^{(n)} - \bar{X}_{\text{old}}^{(n)}) (X_i^{(n)} - \bar{X}_{\text{old}}^{(n)})^T. \quad (21)$$

When the new sample is added, the mean tensor is

$$\begin{aligned} \bar{X} &= \frac{1}{K+1} \sum_{i=1}^{K+1} X_i \\ &= \frac{1}{K+1} \left( \sum_{i=1}^K X_i + X_{\text{new}} \right) = \frac{1}{K+1} (K\bar{X}_{\text{old}} + X_{\text{new}}). \end{aligned} \quad (22)$$

The mode- $n$  covariance matrix is expressed as follows:

$$\begin{aligned} C^{(n)} &= \sum_{i=1}^{K+1} (X_i^{(n)} - \bar{X}^{(n)}) (X_i^{(n)} - \bar{X}^{(n)})^T \\ &= \sum_{i=1}^K (X_i^{(n)} - \bar{X}^{(n)}) (X_i^{(n)} - \bar{X}^{(n)})^T \\ &\quad + (X_{\text{new}}^{(n)} - \bar{X}^{(n)}) (X_{\text{new}}^{(n)} - \bar{X}^{(n)})^T, \end{aligned} \quad (23)$$

where the first item of (23) is

$$\begin{aligned} & \sum_{i=1}^K (X_i^{(n)} - \bar{X}^{(n)}) (X_i^{(n)} - \bar{X}^{(n)})^T \\ &= \sum_{i=1}^K (X_i^{(n)} - \bar{X}_{\text{old}}^{(n)} + \bar{X}_{\text{old}}^{(n)} - \bar{X}^{(n)}) \\ &\quad \times (X_i^{(n)} - \bar{X}_{\text{old}}^{(n)} + \bar{X}_{\text{old}}^{(n)} - \bar{X}^{(n)})^T \\ &= \sum_{i=1}^K \left[ (X_i^{(n)} - \bar{X}_{\text{old}}^{(n)}) + (\bar{X}_{\text{old}}^{(n)} - \bar{X}^{(n)}) \right] \\ &\quad \times \left[ (X_i^{(n)} - \bar{X}_{\text{old}}^{(n)})^T + (\bar{X}_{\text{old}}^{(n)} - \bar{X}^{(n)})^T \right] \\ &= \sum_{i=1}^K (X_i^{(n)} - \bar{X}_{\text{old}}^{(n)}) (X_i^{(n)} - \bar{X}_{\text{old}}^{(n)})^T \\ &\quad + K (\bar{X}_{\text{old}}^{(n)} - \bar{X}^{(n)}) (\bar{X}_{\text{old}}^{(n)} - \bar{X}^{(n)})^T \\ &\quad + \sum_{i=1}^K (X_i^{(n)} - \bar{X}_{\text{old}}^{(n)}) (\bar{X}_{\text{old}}^{(n)} - \bar{X}^{(n)})^T \\ &\quad + (\bar{X}_{\text{old}}^{(n)} - \bar{X}^{(n)}) (X_i^{(n)} - \bar{X}_{\text{old}}^{(n)})^T \\ &= \sum_{i=1}^K (X_i^{(n)} - \bar{X}_{\text{old}}^{(n)}) (X_i^{(n)} - \bar{X}_{\text{old}}^{(n)})^T \\ &\quad + K (\bar{X}_{\text{old}}^{(n)} - \bar{X}^{(n)}) (\bar{X}_{\text{old}}^{(n)} - \bar{X}^{(n)})^T \\ &= C_{\text{old}}^{(n)} + K \left( \bar{X}_{\text{old}}^{(n)} - \frac{K\bar{X}_{\text{old}}^{(n)} + X_{\text{new}}^{(n)}}{K+1} \right) \\ &\quad \times \left( \bar{X}_{\text{old}}^{(n)} - \frac{K\bar{X}_{\text{old}}^{(n)} + X_{\text{new}}^{(n)}}{K+1} \right)^T \\ &= C_{\text{old}}^{(n)} + \frac{K}{(K+1)^2} (\bar{X}_{\text{old}}^{(n)} - X_{\text{new}}^{(n)}) (\bar{X}_{\text{old}}^{(n)} - X_{\text{new}}^{(n)})^T. \end{aligned} \quad (24)$$

The second item of (23) is

$$\begin{aligned} & (X_{\text{new}}^{(n)} - \bar{X}^{(n)}) (X_{\text{new}}^{(n)} - \bar{X}^{(n)})^T \\ &= (X_{\text{new}}^{(n)} - \bar{X}^{(n)}) (X_{\text{new}}^{(n)} - \bar{X}^{(n)})^T \\ &= \left( X_{\text{new}}^{(n)} - \frac{K\bar{X}_{\text{old}}^{(n)} + X_{\text{new}}^{(n)}}{K+1} \right) \left( X_{\text{new}}^{(n)} - \frac{K\bar{X}_{\text{old}}^{(n)} + X_{\text{new}}^{(n)}}{K+1} \right)^T \\ &= \frac{K^2}{(K+1)^2} (\bar{X}_{\text{old}}^{(n)} - X_{\text{new}}^{(n)}) (\bar{X}_{\text{old}}^{(n)} - X_{\text{new}}^{(n)})^T. \end{aligned} \quad (25)$$

Consequently, the mode- $n$  covariance matrix is updated as follows:

$$C^{(n)} = C_{\text{old}}^{(n)} + \frac{K}{K+1} \left( \bar{X}_{\text{old}}^{(n)} - X_{\text{new}}^{(n)} \right) \left( \bar{X}_{\text{old}}^{(n)} - X_{\text{new}}^{(n)} \right)^T. \quad (26)$$

Therefore, when a new sample is added, the projective matrices are solved according to the eigen decomposition on (26).

**3.2. Incremental Learning Based on Multiple Samples.** Given an initial training dataset  $X_{\text{old}} = \{X_1, \dots, X_K\}$ ,  $X_i \in \mathbb{R}^{I_1 \times \dots \times I_N}$ , when new samples are added into training dataset,  $X_{\text{new}} = \{X_{K+1}, \dots, X_{K+T}\}$ , then training dataset becomes into  $\bar{X} = \{X_1, \dots, X_K, X_{K+1}, \dots, X_{K+T}\}$ . In this case, the mean tensor is updated into the following:

$$\begin{aligned} \bar{X} &= \frac{1}{K+T} \sum_{i=1}^{K+T} X_i = \frac{1}{K+T} \left( \sum_{i=1}^K X_i + \sum_{i=K+1}^{K+T} X_i \right) \\ &= \frac{1}{K+T} \left( K\bar{X}_{\text{old}} + T\bar{X}_{\text{new}} \right). \end{aligned} \quad (27)$$

Its mode- $n$  covariance matrix is

$$\begin{aligned} C^{(n)} &= \sum_{i=1}^{K+T} \left( X_i^{(n)} - \bar{X}^{(n)} \right) \left( X_i^{(n)} - \bar{X}^{(n)} \right)^T \\ &= \sum_{i=1}^K \left( X_i^{(n)} - \bar{X}^{(n)} \right) \left( X_i^{(n)} - \bar{X}^{(n)} \right)^T \\ &\quad + \sum_{i=K+1}^{K+T} \left( X_i^{(n)} - \bar{X}^{(n)} \right) \left( X_i^{(n)} - \bar{X}^{(n)} \right)^T. \end{aligned} \quad (28)$$

The first item in (28) is written as follows:

$$\begin{aligned} &\sum_{i=1}^K \left( X_i^{(n)} - \bar{X}^{(n)} \right) \left( X_i^{(n)} - \bar{X}^{(n)} \right)^T \\ &= \sum_{i=1}^K \left( X_i^{(n)} - \bar{X}_{\text{old}}^{(n)} \right) \left( X_i^{(n)} - \bar{X}_{\text{old}}^{(n)} \right)^T \\ &\quad + K \left( \bar{X}_{\text{old}}^{(n)} - \bar{X}^{(n)} \right) \left( \bar{X}_{\text{old}}^{(n)} - \bar{X}^{(n)} \right)^T \\ &\quad + \sum_{i=1}^K \left[ \left( X_i^{(n)} - \bar{X}_{\text{old}}^{(n)} \right) \left( \bar{X}_{\text{old}}^{(n)} - \bar{X}^{(n)} \right)^T \right. \\ &\quad \left. + \left( \bar{X}_{\text{old}}^{(n)} - \bar{X}^{(n)} \right) \left( X_i^{(n)} - \bar{X}_{\text{old}}^{(n)} \right)^T \right], \end{aligned} \quad (29)$$

where

$$\begin{aligned} &\sum_{i=1}^K \left[ \left( X_i^{(n)} - \bar{X}_{\text{old}}^{(n)} \right) \left( \bar{X}_{\text{old}}^{(n)} - \bar{X}^{(n)} \right)^T \right. \\ &\quad \left. + \left( \bar{X}_{\text{old}}^{(n)} - \bar{X}^{(n)} \right) \left( X_i^{(n)} - \bar{X}_{\text{old}}^{(n)} \right)^T \right] \\ &= K\bar{X}_{\text{old}}^{(n)}\bar{X}_{\text{old}}^{(n)T} - K\bar{X}_{\text{old}}^{(n)}\bar{X}^{(n)T} - K\bar{X}_{\text{old}}^{(n)}\bar{X}_{\text{old}}^{(n)T} \\ &\quad + K\bar{X}_{\text{old}}^{(n)}\bar{X}^{(n)T} + K\bar{X}_{\text{old}}^{(n)}\bar{X}_{\text{old}}^{(n)T} - K\bar{X}_{\text{old}}^{(n)}\bar{X}_{\text{old}}^{(n)T} \\ &\quad - K\bar{X}^{(n)}\bar{X}_{\text{old}}^{(n)T} + K\bar{X}^{(n)}\bar{X}_{\text{old}}^{(n)T} \\ &= 0, \\ &K \left( \bar{X}_{\text{old}}^{(n)} - \bar{X}^{(n)} \right) \left( \bar{X}_{\text{old}}^{(n)} - \bar{X}^{(n)} \right)^T \\ &= \frac{KT^2}{(K+T)^2} \left( \bar{X}_{\text{old}}^{(n)} - \bar{X}_{\text{new}}^{(n)} \right) \left( \bar{X}_{\text{old}}^{(n)} - \bar{X}_{\text{new}}^{(n)} \right)^T. \end{aligned} \quad (30)$$

Putting (30) into (29), then (29) becomes as follows:

$$\begin{aligned} &\sum_{i=1}^K \left( X_i^{(n)} - \bar{X}^{(n)} \right) \left( X_i^{(n)} - \bar{X}^{(n)} \right)^T \\ &= C_{\text{old}}^{(n)} + \frac{KT^2}{(K+T)^2} \left( \bar{X}_{\text{old}}^{(n)} - \bar{X}_{\text{new}}^{(n)} \right) \left( \bar{X}_{\text{old}}^{(n)} - \bar{X}_{\text{new}}^{(n)} \right)^T. \end{aligned} \quad (31)$$

The second item in (28) is written as follows:

$$\begin{aligned} &\sum_{i=K+1}^{K+T} \left( X_i^{(n)} - \bar{X}^{(n)} \right) \left( X_i^{(n)} - \bar{X}^{(n)} \right)^T \\ &= C_{\text{new}}^{(n)} + T \left( \bar{X}_{\text{new}}^{(n)} - \bar{X}^{(n)} \right) \left( \bar{X}_{\text{new}}^{(n)} - \bar{X}^{(n)} \right)^T, \end{aligned} \quad (32)$$

where

$$\begin{aligned} &T \left( \bar{X}_{\text{new}}^{(n)} - \bar{X}^{(n)} \right) \left( \bar{X}_{\text{new}}^{(n)} - \bar{X}^{(n)} \right)^T \\ &= \frac{K^2T}{(K+T)^2} \left( \bar{X}_{\text{old}}^{(n)} - \bar{X}_{\text{new}}^{(n)} \right) \left( \bar{X}_{\text{old}}^{(n)} - \bar{X}_{\text{new}}^{(n)} \right)^T \end{aligned} \quad (33)$$

Then (32) becomes as follows:

$$\begin{aligned} &\sum_{i=K+1}^{K+T} \left( X_i^{(n)} - \bar{X}^{(n)} \right) \left( X_i^{(n)} - \bar{X}^{(n)} \right)^T \\ &= C_{\text{new}}^{(n)} + \frac{K^2T}{(K+T)^2} \left( \bar{X}_{\text{old}}^{(n)} - \bar{X}_{\text{new}}^{(n)} \right) \left( \bar{X}_{\text{old}}^{(n)} - \bar{X}_{\text{new}}^{(n)} \right)^T \end{aligned} \quad (34)$$

Putting (31) and (34) into (28), then we can get the following:

$$C^{(n)} = C_{\text{old}}^{(n)} + C_{\text{new}}^{(n)} + \frac{KT}{K+T} \left( \bar{X}_{\text{old}}^{(n)} - \bar{X}_{\text{new}}^{(n)} \right) \left( \bar{X}_{\text{old}}^{(n)} - \bar{X}_{\text{new}}^{(n)} \right)^T. \quad (35)$$

It is worthy to note that when new samples are available, it has no need to recompute the mode- $n$  covariance matrix of all training samples. We just have to solve the mode- $n$  covariance matrix of new added samples and the difference between original training samples and new added samples. However, like traditional incremental PCA, eigen decomposition on  $C^{(n)}$  has been repeated once new samples are added. It is certain that the repeated eigen decomposition on  $C^{(n)}$  will cause heavy computational cost, which is called “the eigen decomposition updating problem.” For traditional vector-based incremental learning algorithm, the updated-SVD technique is proposed in [25] to fit the eigen decomposition. This paper will introduce the updated-SVD technique into tensor-based incremental learning algorithm.

For original samples, the mode- $n$  covariance matrix is

$$C_{\text{old}}^{(n)} = \sum_{i=1}^K (X_i^{(n)} - \bar{X}_{\text{old}}^{(n)}) (X_i^{(n)} - \bar{X}_{\text{old}}^{(n)})^T = S_{\text{old}}^{(n)} S_{\text{old}}^{(n)T}, \quad (36)$$

where  $S_{\text{old}}^{(n)} = [X_1^{(n)} - \bar{X}_{\text{old}}^{(n)}, \dots, X_K^{(n)} - \bar{X}_{\text{old}}^{(n)}]$ . According to the eigen decomposition  $S_{\text{old}}^{(n)} = \text{svd}(U\Sigma V^T)$ , we can get the following:

$$\begin{aligned} S_{\text{old}}^{(n)} S_{\text{old}}^{(n)T} &= (U\Sigma V^T) (U\Sigma V^T)^T \\ &= U\Sigma V^T V\Sigma U^T = U\Sigma^2 U^T = \text{eig}(C_{\text{old}}^{(n)}). \end{aligned} \quad (37)$$

So it is easy to derive that the eigen-vector of  $C_{\text{old}}^{(n)}$  is the left singular vector of  $S_{\text{old}}^{(n)}$  and the eigen-values correspond to the extraction of left singular values of  $S_{\text{old}}^{(n)}$ .

For new samples, the mode- $n$  covariance matrix is

$$C_{\text{new}}^{(n)} = \sum_{i=K+1}^{K+T} (X_i^{(n)} - \bar{X}_{\text{new}}^{(n)}) (X_i^{(n)} - \bar{X}_{\text{new}}^{(n)})^T = S_{\text{new}}^{(n)} S_{\text{new}}^{(n)T}, \quad (38)$$

where  $S_{\text{new}}^{(n)} = [X_1^{(n)} - \bar{X}_{\text{new}}^{(n)}, \dots, X_K^{(n)} - \bar{X}_{\text{new}}^{(n)}]$ . According to (35), the updated mode- $n$  covariance matrix is defined as follows:

$$\begin{aligned} C^{(n)} &= C_{\text{old}}^{(n)} + C_{\text{new}}^{(n)} + \frac{KT}{K+T} \\ &\times (\bar{X}_{\text{old}}^{(n)} - \bar{X}_{\text{new}}^{(n)}) (\bar{X}_{\text{old}}^{(n)} - \bar{X}_{\text{new}}^{(n)})^T = S^{(n)} S^{(n)T}, \end{aligned} \quad (39)$$

where  $S^{(n)} = [S_{\text{old}}^{(n)}, S_{\text{new}}^{(n)}, \sqrt{KT/(K+T)}(\bar{X}_{\text{old}}^{(n)} - \bar{X}_{\text{new}}^{(n)})]$ . Therefore, the updated projective matrix  $U^{(n)}$  is the eigen-vectors corresponding to the largest  $P_n$  eigen-values of  $S^{(n)}$ . The main steps of incremental tensor principal component analysis are listed as follows:

input: original samples and new added samples,

output:  $N$  projective matrices.

*Step 1.* Computing and saving

$$\text{eig}(C_{\text{old}}^{(n)}) \approx [U_r^{(n)}, \Sigma_r^{(n)}]. \quad (40)$$

*Step 2.* For  $i = 1 : N$

$$B = \left[ S_{\text{new}}^{(n)}, \sqrt{\frac{KT}{K+T}} (\bar{X}_{\text{old}}^{(n)} - \bar{X}_{\text{new}}^{(n)}) \right]. \quad (41)$$

Processing QR decomposition for the following equation:

$$QR = (I - U_r^{(n)} U_r^{(n)T}) B. \quad (42)$$

Processing SVD decomposition for the following equation:

$$\text{svd} \left[ \begin{array}{c} \sqrt{\Sigma_r^{(n)}} U_r^{(n)T} B \\ 0 \quad R \end{array} \right] = \widehat{U} \widehat{\Sigma} \widehat{V}^T. \quad (43)$$

Computing the following equation:

$$[S_{\text{old}}^{(n)}, B] \approx ([U_r^{(n)}, Q] \widehat{U}) \widehat{\Sigma} \left( \begin{bmatrix} V_r^{(n)} & 0 \\ 0 & I \end{bmatrix} \widehat{V} \right)^T. \quad (44)$$

Then the updated projective matrix is computed as follows:

$$U^{(n)} = [U_r^{(n)}, Q] \widehat{U}, \quad (45)$$

end.

*Step 3.* Repeating the above steps until the incremental learning is finished.

**3.3. The Complexity Analysis.** For tensor dataset  $X = \{X_1, \dots, X_M\}$ ,  $X_i \in \mathbb{R}^{I_1 \times \dots \times I_N}$ , without loss of generality, it is assumed that all dimensions are equal, that is,  $I_1 = \dots = I_N = I$ .

Vector-based PCA converts all data into vector and constructs a data matrix  $X \in \mathbb{R}^{M \times D}$ ,  $D = I^N$ . For vector-based PCA, the main computational cost contains three parts: the computation of the covariance matrix, the eigen decomposition of the covariance matrix, and the computation of low-dimensional features. The time complexity to compute covariance matrix is  $O(MI^{2N})$ , the time complexity of the eigen decomposition is  $O(I^{3N})$ , and the time complexity to compute low-dimensional features is  $O(MI^{2N} + I^{3N})$ .

Letting the iterative number be 1, the time complexity to computing the mode- $n$  covariance matrix for MPCA is  $O(MNI^{N+1})$ , the time complexity of eigen decomposition is  $O(NI^3)$ , and the time complexity to compute low-dimensional features is  $O(MNI^{N+1})$ , so the total time complexity is  $O(MNI^{N+1} + NI^3)$ . Considering the time complexity, MPCA is superior to PCA.

For ITPCA, it is assumed that  $T$  incremental datasets are added; MPCA has to recompute mode- $n$  covariance matrix and conducts eigen decomposition for initial dataset and incremental dataset. The more the training samples are, the higher time complexity they have. If updated-SVD is used, we only need to compute QR decomposition and SVD decomposition. The time complexity of QR decomposition is  $O(NI^{N+1})$ . The time complexity of the rank- $k$  decomposition of the matrix with the size of  $(r+I) \times (r+I^{N-1})$  is



FIGURE 1: The samples in USPS dataset.

$O(N(r + I)k^2)$ . It can be seen that the time complexity of updated-SVD has nothing to do with the number of new added samples.

Taking the space complexity into account, if training samples are reduced into low-dimensional space and the dimension is  $D = \prod_{n=1}^N d_n$ , then PCA needs  $D \prod_{n=1}^N I_n$  bytes to save projective matrices and MPCA needs  $\sum_{n=1}^N I_n d_n$  bytes. So MPCA has lower space complexity than PCA. For incremental learning, both PCA and MPCA need  $M \prod_{n=1}^N I_n$  bytes to save initial training samples; ITPCA only need  $\sum_{n=1}^N I_n^2$  bytes to keep mode- $n$  covariance matrix.

### 4. Experiments

In this section, the handwritten digit recognition experiments on the USPS image dataset are conducted to evaluate the performance of incremental tensor principal component analysis. The USPS handwritten digit dataset has 9298 images from zero to nine shown in Figure 1. For each image, the size is  $16 \times 16$ . In this paper, we choose 1000 images and divide them into initial training samples, new added samples, and test samples. Furthermore, the nearest neighbor classifier is employed to classify the low-dimensional features. The recognition results are compared with PCA [26], IPCA [15], and MPCA [11].

At first, we choose 70 samples belonging to four classes from initial training samples. For each time of incremental learning, 70 samples which belong to the other two classes are added. So after three times, the class labels of the training samples are ten and there are 70 samples in each class. The resting samples of original training samples are considered as testing dataset. All algorithms are implemented in MATLAB 2010 on an Intel (R) Core (TM) i5-3210 M CPU @ 2.5 GHz with 4 G RAM.

Firstly, 36 PCs are preserved and fed into the nearest neighbor classifier to obtain the recognition results. The results are plotted in Figure 2. It can be seen that MPCA and ITPCA are better than PCA and IPCA for initial learning; the probable reason is that MPCA and ITPCA employ tensor representation to preserve the structure information.

The recognition results under different learning stages are shown in Figures 3, 4, and 5. It can be seen that the recognition results of these four methods always fluctuate violently

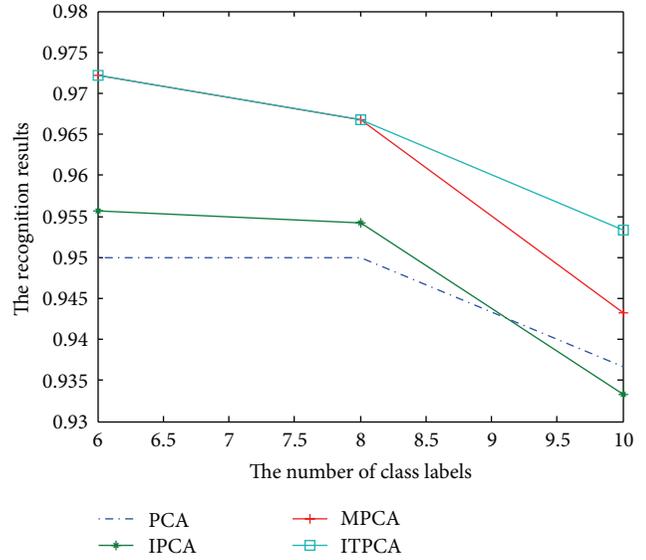


FIGURE 2: The recognition results for 36 PCs of the initial learning.

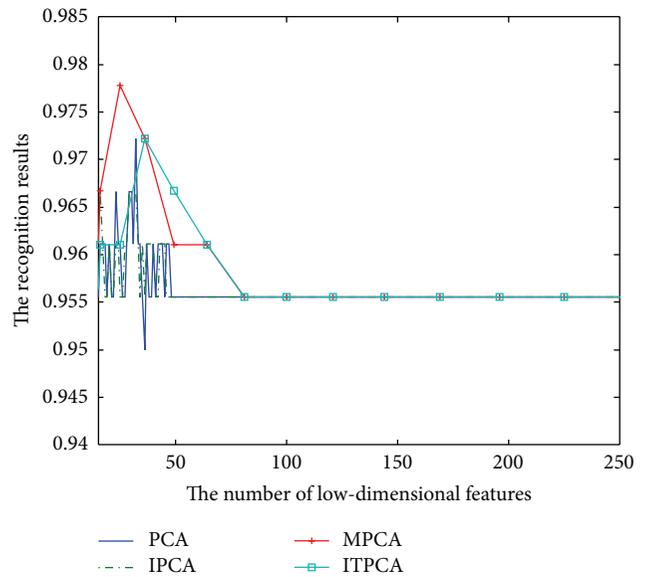


FIGURE 3: The recognition results of different methods of the first incremental learning.

when the numbers of low-dimensional features are small. However, with the increment of the feature number, the recognition performance keeps stable. Generally MPCA and ITPCA are superior to PCA and IPCA. Although ITPCA have comparative performance at first two learning, ITPCA begin to surmount MPCA after the third learning. Figure 6 has given the best recognition percents of different methods. We can get the same conclusion as shown in Figures 3, 4, and 5.

The time and space complexity of different methods are shown in Figures 7 and 8, respectively. Taking the time complexity into account, it can be found that at the stage of initial learning, PCA has the lowest time complexity. With

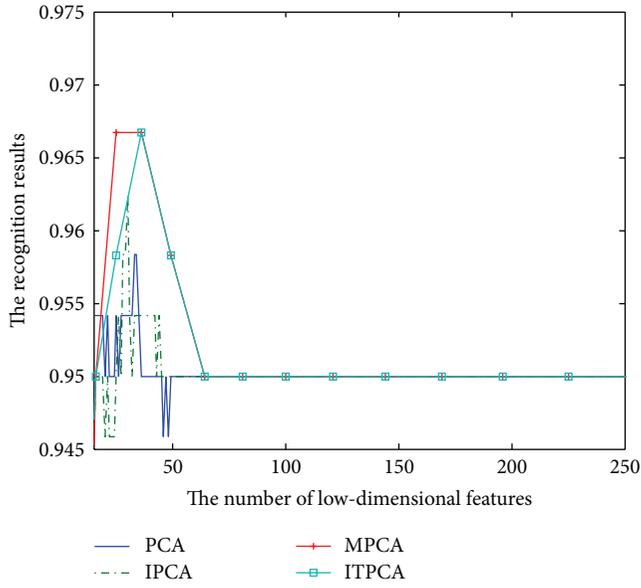


FIGURE 4: The recognition results of different methods of the second incremental learning.

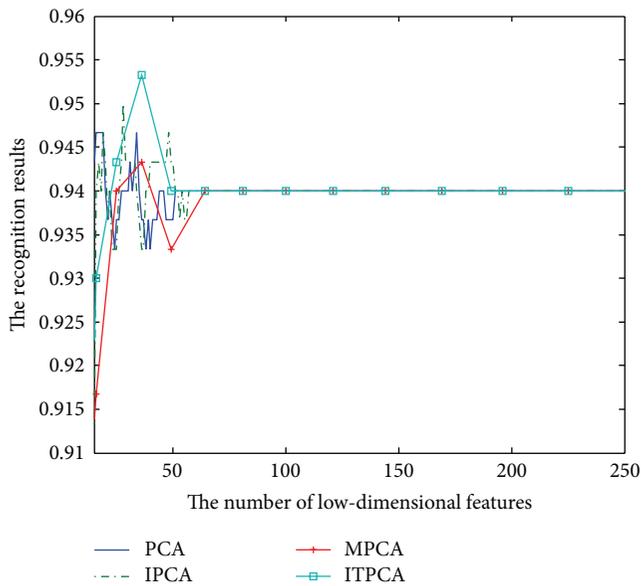


FIGURE 5: The recognition results of different methods of the third incremental learning.

the increment of new samples, the time complexity of PCA and MPCA grows greatly and the time complexity of IPCA and ITPCA becomes stable. ITPCA has slower increment than MPCA. The reason is that ITPCA introduces incremental learning based on the updated-SVD technique and avoids decomposing the mode- $n$  covariance matrix of original samples again. Considering the space complexity, it is easy to find that ITPCA has the lowest space complexity among four compared methods.

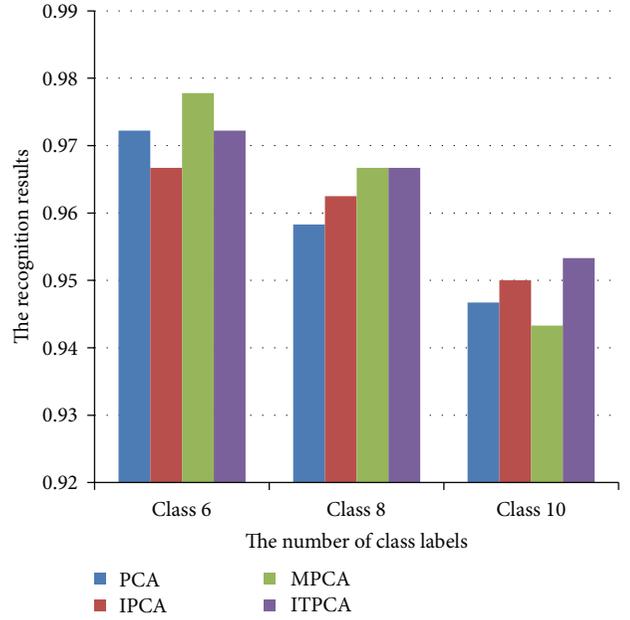


FIGURE 6: The comparison of recognition performance of different methods.

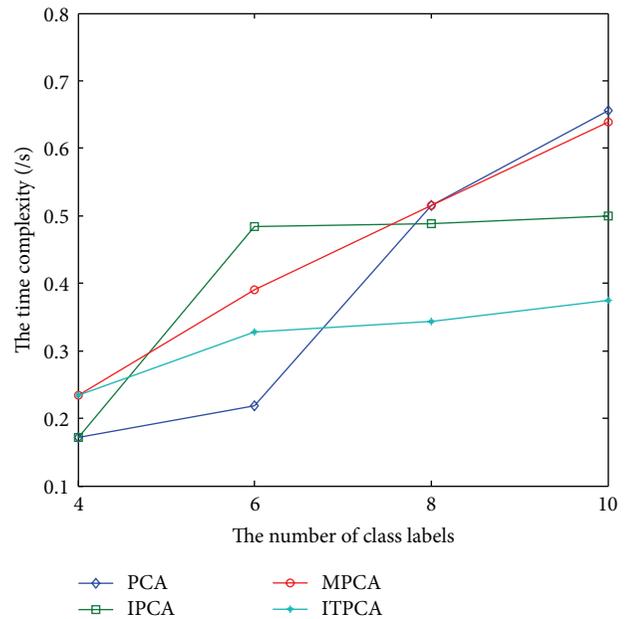


FIGURE 7: The comparison of time complexity of different methods.

## 5. Conclusion

This paper presents incremental tensor principal component analysis based on updated-SVD technique to take full advantage of redundancy of the space structure information and online learning. Furthermore, this paper proves that PCA and 2DPCA are the special cases of MPCA and all of them can be unified into the graph embedding framework. This

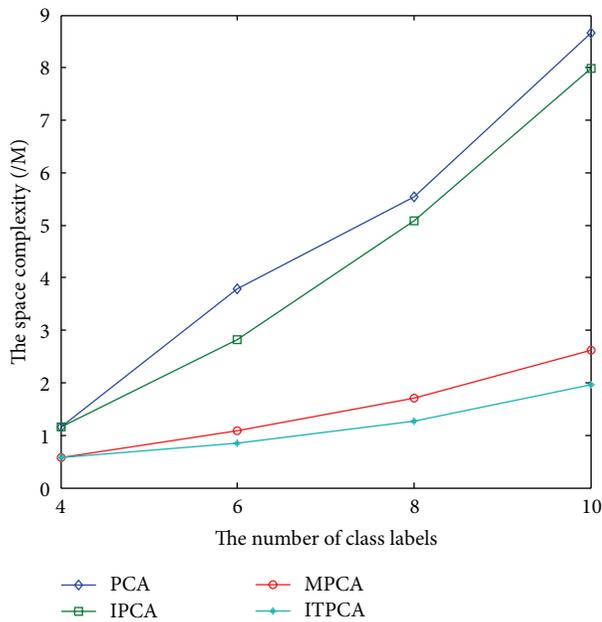


FIGURE 8: The comparison of space complexity of different methods.

paper also analyzes incremental learning based on single sample and multiple samples in detail. The experiments on handwritten digit recognition have demonstrated that principal component analysis based on tensor representation is superior to tensor principal component analysis based on vector representation. Although at the stage of initial learning, MPCA has better recognition performance than ITPCA, the learning capability of ITPCA becomes well gradually and exceeds MPCA. Moreover, even if new samples are added, the time and space complexity of ITPCA still keep slower increment.

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## Acknowledgments

This present work has been funded with support from the National Natural Science Foundation of China (61272448), the Doctoral Fund of Ministry of Education of China (20110181130007), the Young Scientist Project of Chengdu University (no. 2013XJZ21).

## References

- [1] H. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos, "Uncorrelated multilinear discriminant analysis with regularization and aggregation for tensor object recognition," *IEEE Transactions on Neural Networks*, vol. 20, no. 1, pp. 103–123, 2009.
- [2] C. Liu, K. He, J.-L. Zhou, and C.-B. Gao, "Discriminant orthogonal rank-one tensor projections for face recognition," in *Intelligent Information and Database Systems*, N. T. Nguyen, C.-G. Kim, and A. Janiak, Eds., vol. 6592 of *Lecture Notes in Computer Science*, pp. 203–211, 2011.
- [3] G.-F. Lu, Z. Lin, and Z. Jin, "Face recognition using discriminant locality preserving projections based on maximum margin criterion," *Pattern Recognition*, vol. 43, no. 10, pp. 3572–3579, 2010.
- [4] D. Tao, X. Li, X. Wu, and S. J. Maybank, "General tensor discriminant analysis and Gabor features for gait recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 10, pp. 1700–1715, 2007.
- [5] F. Nie, S. Xiang, Y. Song, and C. Zhang, "Extracting the optimal dimensionality for local tensor discriminant analysis," *Pattern Recognition*, vol. 42, no. 1, pp. 105–114, 2009.
- [6] Z.-Z. Yu, C.-C. Jia, W. Pang, C.-Y. Zhang, and L.-H. Zhong, "Tensor discriminant analysis with multiscale features for action modeling and categorization," *IEEE Signal Processing Letters*, vol. 19, no. 2, pp. 95–98, 2012.
- [7] S. J. Wang, J. Yang, M. F. Sun, X. J. Peng, M. M. Sun, and C. G. Zhou, "Sparse tensor discriminant color space for face verification," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 6, pp. 876–888, 2012.
- [8] J. L. Minoi, C. E. Thomaz, and D. F. Gillies, "Tensor-based multivariate statistical discriminant methods for face applications," in *Proceedings of the International Conference on Statistics in Science, Business, and Engineering (ICSSBE '12)*, pp. 1–6, September 2012.
- [9] N. Tang, X. Gao, and X. Li, "Tensor subclass discriminant analysis for radar target classification," *Electronics Letters*, vol. 48, no. 8, pp. 455–456, 2012.
- [10] H. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos, "A survey of multilinear subspace learning for tensor data," *Pattern Recognition*, vol. 44, no. 7, pp. 1540–1551, 2011.
- [11] H. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos, "MPCA: multilinear principal component analysis of tensor objects," *IEEE Transactions on Neural Networks*, vol. 19, no. 1, pp. 18–39, 2008.
- [12] S. Yan, D. Xu, B. Zhang, H.-J. Zhang, Q. Yang, and S. Lin, "Graph embedding and extensions: a general framework for dimensionality reduction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 1, pp. 40–51, 2007.
- [13] R. Plamondon and S. N. Srihari, "On-line and off-line handwriting recognition: a comprehensive survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 1, pp. 63–84, 2000.
- [14] C. M. Johnson, "A survey of current research on online communities of practice," *Internet and Higher Education*, vol. 4, no. 1, pp. 45–60, 2001.
- [15] P. Hall, D. Marshall, and R. Martin, "Merging and splitting eigenspace models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 9, pp. 1042–1049, 2000.
- [16] J. Sun, D. Tao, S. Papadimitriou, P. S. Yu, and C. Faloutsos, "Incremental tensor analysis: theory and applications," *ACM Transactions on Knowledge Discovery from Data*, vol. 2, no. 3, article 11, 2008.
- [17] J. Wen, X. Gao, Y. Yuan, D. Tao, and J. Li, "Incremental tensor biased discriminant analysis: a new color-based visual tracking method," *Neurocomputing*, vol. 73, no. 4–6, pp. 827–839, 2010.
- [18] J.-G. Wang, E. Sung, and W.-Y. Yau, "Incremental two-dimensional linear discriminant analysis with applications to face recognition," *Journal of Network and Computer Applications*, vol. 33, no. 3, pp. 314–322, 2010.

- [19] X. Qiao, R. Xu, Y.-W. Chen, T. Igarashi, K. Nakao, and A. Kashimoto, "Generalized N-Dimensional Principal Component Analysis (GND-PCA) based statistical appearance modeling of facial images with multiple modes," *IPSJ Transactions on Computer Vision and Applications*, vol. 1, pp. 231–241, 2009.
- [20] H. Kong, X. Li, L. Wang, E. K. Teoh, J.-G. Wang, and R. Venkateswarlu, "Generalized 2D principal component analysis," in *Proceedings of the IEEE International Joint Conference on Neural Networks (IJCNN '05)*, vol. 1, pp. 108–113, August 2005.
- [21] D. Zhang and Z.-H. Zhou, "(2D)2 PCA: two-directional two-dimensional PCA for efficient face representation and recognition," *Neurocomputing*, vol. 69, no. 1–3, pp. 224–231, 2005.
- [22] J. Ye, "Generalized low rank approximations of matrices," *Machine Learning*, vol. 61, no. 1–3, pp. 167–191, 2005.
- [23] J. Yang, D. Zhang, A. F. Frangi, and J.-Y. Yang, "Two-dimensional PCA: a new approach to appearance-based face representation and recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 1, pp. 131–137, 2004.
- [24] J. Yang and J.-Y. Yang, "From image vector to matrix: a straightforward image projection technique-IMPCA vs. PCA," *Pattern Recognition*, vol. 35, no. 9, pp. 1997–1999, 2002.
- [25] J. Kwok and H. Zhao, "Incremental eigen decomposition," in *Proceedings of the International Conference on Artificial Neural Networks (ICANN '03)*, pp. 270–273, Istanbul, Turkey, June 2003.
- [26] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. fisherfaces: recognition using class specific linear projection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 711–720, 1997.

## Research Article

# Structural Reliability Assessment by Integrating Sensitivity Analysis and Support Vector Machine

Shao-Fei Jiang,<sup>1</sup> Da-Bao Fu,<sup>2</sup> and Si-Yao Wu<sup>1</sup>

<sup>1</sup> College of Civil Engineering, Fuzhou University, Fuzhou 350108, China

<sup>2</sup> Fuzhou Planning & Design Research Institute, Fuzhou 350003, China

Correspondence should be addressed to Shao-Fei Jiang; cejsf@163.com

Received 1 September 2013; Accepted 23 December 2013; Published 23 January 2014

Academic Editor: Orwa Jaber Housheya

Copyright © 2014 Shao-Fei Jiang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

To reduce the runtime and ensure enough computation accuracy, this paper proposes a structural reliability assessment method by the use of sensitivity analysis (SA) and support vector machine (SVM). The sensitivity analysis is firstly applied to assess the effect of random variables on the values of performance function, while the small-influence variables are rejected as input vectors of SVM. Then, the trained SVM is used to classify the input vectors, which are produced by sampling the residual variables based on their distributions. Finally, the reliability assessment is implemented with the aid of reliability theory. A 10-bar planar truss is used to validate the feasibility and efficiency of the proposed method, and a performance comparison is made with other existing methods. The results show that the proposed method can largely save the runtime with less reduction of the accuracy; furthermore, the accuracy using the proposed method is the highest among the methods employed.

## 1. Introduction

In recent years, a number of structural reliability assessment methods, including first-order reliability method (FORM) [1], response surface method (RSM) [2], and Monte-Carlo simulation method (MCSM) [3], have been developed and applied to practical engineering structures. Among these methods, the FORM is usually used to directly estimate structural failure probability in the case of the explicit limit state functions. In contrast, the RSM and MCSM are widely used in the case that the limit state functions are complex and implicit. The main idea of RSM is to transfer the original implicit limit state function to an approximated explicit expression, which will then be used for the assessment of failure probability with the aid of FORM. However, in most cases, the hypothetical explicit expressions can hardly be found to represent precisely the original nonlinear and complex functions; thus RSM usually causes an unallowable error, even a wrong assessment result. MCSM not only is the most precise method for failure probability assessment, but also solves theoretically all of reliability problems. However, MCSM is a time-consuming process. It is suitable for solving such problem when structural failure probability is small,

because a number of samples are required for the purpose of obtaining a reasonable result.

To overcome the low-fidelity of RSM and low computing efficiency of MCSM, several researchers have attempted to construct the limit state function based on the intellectual techniques, such as artificial neural networks [4, 5] and SVM [6, 7]. Due to the strong small-samples learning ability and generalization capability of SVM [8, 9], it has been widely used for structural reliability analysis in various fields. Hurtado and Alvarez [10] regarded reliability problems as model classification problems and combined the SVM and FEA for the assessment of structural failure probability. Hurtado [11] used the statistical learning theory to prove the feasibility of SVM in the application of reliability problems. Jin et al. [12] combined RSM with SVM for structural reliability probability assessment, and the results showed that this method is more accurate and efficient in comparison with other conventional methods. Guo and Bai [13] introduced the least squares SVM for regression into reliability analysis to deal with huge computational cost and huge space demands.

The input vectors of SVM model are the variables influencing the structural reliability assessment. For a large-scale civil structure, its reliability is affected by a number

of variables due to the complex service environment and loading situations. If all of influencing variables are taken into consideration with no regard to their importance in the process of reliability assessment, it will increase the sample size of input variables, complicate the SVM model, and enlarge the data storage memory demands while decreasing the classification accuracy (CA) of SVM model. In fact, some input variables have slight effect on the reliability assessment results. Therefore, it is necessary to eliminate the small-influencing variables before assessing the structure reliability. Recently, a series of SA techniques have been developed and studied for the purpose of quantifying the importance of input variables. These SA techniques are divided into two classes: global SA methods and local SA methods [14]. The local SA methods are usually based on differential and/or difference theory and ignore the probability distributions of variables, thus not competent for analyzing the random variables in limit state function. The global SA methods consider not only the effects of the probability distributions of individual input variables on the output, but also the contribution of the interaction among input variables on the output. In the past decades, Sobol's SA method [15–20], as an invaluable tool, has drawn researchers' attention because it works well without simplifying approximations, even for the functions with large number of variables.

In order to reduce the dimension of input samples, simplifying the SVM model in the case of ensuring computation accuracy, this paper presents a novelty reliability analysis method based on SA and SVM. The small-influence variables in the limit state function are extracted in virtue of Sobol's SA method and are rejected as input vectors of SVM model. The SVM model is trained and tested by samples of residual variables. The reliability assessment is implemented with the aid of reliability theory. To validate the applicability and efficiency of the proposed method, the reliability assessment of a 10-bar planar truss is employed. In addition, some comparisons are also carried out.

## 2. Structural Reliability Assessment Methodologies

**2.1. Sobol's Global SA Method.** Sobol's method is a variance-based global SA technique that has been applied to assess the relative importance of input variables on the output. It is able to decompose the variance of the output into terms due to individual input variables and terms due to the interactions between input variables.

Consider a square integrable function,  $f(\mathbf{x}) = f(x_1, x_2, \dots, x_n)$ , as a function of vector of input variables  $x_j$ , where  $\mathbf{x} \in \Omega^n$  is the  $n$ -dimensional unit hypercube. If the input variables are mutually independent then there exists an interesting decomposition of  $f(\mathbf{x})$ :

$$f(\mathbf{x}) = f_0 + \sum_{i=1}^n f_i(x_i) + \sum_{1 < i < j < n} f_{ij}(x_i, x_j) + \dots + f_{1, \dots, n}(x_1, \dots, x_n), \quad (1)$$

where  $f_0$  is a constant. The total number of summands in (1) is  $2^n$ .

If the following condition

$$\int_0^1 f_{i_1, \dots, i_s} dx_k = 0 \quad (2)$$

is imposed for  $k = i_1, \dots, i_s$ , where  $1 \leq i_1 \leq \dots \leq i_s \leq n$ , the decomposition described in (1) is unique. Moreover, all of summands are mutually orthogonal and can be obtained with the aid of multiple integral:

$$f_0 = \int_{\Omega^n} f(\mathbf{x}) d\mathbf{x} \quad (3)$$

$$f_i(x_i) = -f_0 + \int_0^1 \dots \int_0^1 f(\mathbf{x}) d\mathbf{x}_{\sim i} \quad (4)$$

$$f_{ij}(x_i, x_j) = -f_0 - f_i(x_i) - f_j(x_j) + \int_0^1 \dots \int_0^1 f(\mathbf{x}) d\mathbf{x}_{\sim (ij)}, \quad (5)$$

where  $\mathbf{x}_{\sim i}$  is the set of input variables except  $x_i$  and  $\mathbf{x}_{\sim (ij)}$  is the set of input variables except  $x_i$  and  $x_j$ . The higher dimensional summands are similarly found except for the last one that is calculated using (1).

Therefore, the partial variances,  $D_{i_1, \dots, i_s}$ , representing the contribution of each of summands to the total variance of output, can be expressed as

$$D_{i_1, \dots, i_s} = \int_0^1 \dots \int_0^1 f_{i_1, \dots, i_s}^2 dx_{i_1} \dots dx_{i_s} \quad (6)$$

with the total variance equal to

$$D = \int_{\Omega^n} f^2(\mathbf{x}) d\mathbf{x} - f_0^2 \quad (7)$$

which can also be expressed as

$$D = \sum_{s=1}^n \sum_{i_1 < \dots < i_s}^n D_{i_1, \dots, i_s}. \quad (8)$$

The relative importance of input variables is quantified by a set of indices, namely, first-order ( $S_{i1}$ ) and total ( $TS_i$ ) sensitivity indices. The former represents the contribution of the individual  $x_i$  on the total variance without any interactions with other input variables, while the latter refers to the contribution of all input variables. In addition, the  $s$ -order ( $s \geq 1$ ) sensitivity index,  $S_{is}$ , represents the coupled contribution of the interaction among  $s$  input variables on the total variance. It is given by

$$S_{is} = \frac{D_{is}}{D} \quad \text{for } is = i_1, \dots, i_s, \quad (9)$$

In order to investigate the total sensitivity index ( $TS_i$ ) of input variables  $x_i$ , the total variance,  $D$ , can be divided into two complementary terms:  $D_i$  and  $D_{\sim i}$ , where  $D_{\sim i}$  denotes the variance due to all of input variables except  $x_i$ . Therefore, the

total sensitivity index (TS<sub>*i*</sub>) of input variable  $x_i$  is expressed as

$$TS_i = 1 - \frac{D_{\sim i}}{D}. \quad (10)$$

The variances in (6) can be calculated approximately by Monte-Carlo numerical integrations, particularly when the function,  $f(\mathbf{x})$ , is highly nonlinear and/or implicit [16, 21]. The Monte-Carlo method approximations for  $f_0$ ,  $D$ , and  $D_{\sim i}$  are given by

$$\hat{f}_0 = \frac{1}{N} \sum_{m=1}^N f(\mathbf{x}_m) \quad (11)$$

$$\hat{D} = \frac{1}{N} \sum_{m=1}^N f^2(\mathbf{x}_m) - \hat{f}_0^2 \quad (12)$$

$$\hat{D}_{\sim i} = \frac{1}{N} \sum_{m=1}^N f(\mathbf{x}_{(\sim i)m}^{(1)}, x_{im}^{(1)}) f(\mathbf{x}_{(\sim i)m}^{(1)}, x_{im}^{(2)}) - \hat{f}_0^2, \quad (13)$$

where  $N$  is the sample size,  $\mathbf{x}_m$  is the samples in the  $n$ -dimensional unit hypercube, and superscripts (1) and (2) represent two different samples, respectively. The  $x_{im}^{(1)}$  and  $x_{im}^{(2)}$  denote that the input variables  $x_{im}$  use the sampled values in samples (1) and (2), respectively. The  $x_{(\sim i)m}^{(1)}$  and  $x_{(\sim i)m}^{(2)}$  represent cases when all the input variables except  $x_{im}$  use the sampled values in samples (1) and (2), respectively.

Usually, the input variables whose total sensitivity indices are less than 0.3 are considered to be slight of contribution on the output of  $f(\mathbf{x})$  [22]. Therefore, it is reasonable to define a threshold value of 0.05 to eliminate the small-influence input variables.

**2.2. SVM.** SVM is an emerging machine learning technique that has been successfully applied to pattern classification and regression analysis. It is based on the Vapnik-Chervonenkis dimension of statistical learning theory and the principle of structural risk minimization; thus it has a better generalization capability than the conventional classification methods. This is based on the principle of empirical risk minimization.

Suppose a set of training examples  $\mathbf{x} = \{\mathbf{x}_l \mid l = 1, \dots, M\}$  are input vectors in space  $\mathbf{x}_l \in \mathbf{R}^d$  with associated labels  $y_l \in \{-1, +1\}$  (-1: label for class I, +1: label for class II). Some kernel functions are used to map the input samples to a high-dimensional feature space so that the overlapped samples in the original space become linearly separable in the feature space. Therefore, there exist hyperplanes separating the positive examples on one side and the negative samples on the other side. The hyperplanes are given by

$$y_l (\mathbf{w}^T K(\mathbf{x}, \mathbf{x}_l) + b) - 1 \geq 0 \quad l = 1, \dots, M, \quad (14)$$

where  $\mathbf{w}$  and  $b$  are the weight vector and bias of hyperplane, respectively.

Among these separating hyperplanes, the one so-called optimal separating hyperplane (OSH) separates all vectors without error and the distance between the closest vectors to

the hyperplane is maximal. The OSH is found by minimizing  $\|\mathbf{w}\|^2$  under constraints of (14). Therefore, the primal form of objective function is

$$\min_{\mathbf{w}, b} \quad \frac{1}{2} \mathbf{w}^T \mathbf{w} \quad (15)$$

$$\text{s.t.} \quad y_l (\mathbf{w}^T K(\mathbf{x}, \mathbf{x}_l) + b) \geq 1.$$

The Lagrange multipliers,  $\alpha_l$ , are employed to solve the above problem. Consequently, the optimal problem is rewritten as a dual form:

$$\max \quad L(\alpha) = \sum_{l=1}^M \alpha_l - \frac{1}{2} \|\mathbf{w}\|^2 = \sum_{l=1}^M \alpha_l - \frac{1}{2} \sum_{l=1}^M \sum_{p=1}^M y_l y_p K(\mathbf{x}_l, \mathbf{x}_p)$$

$$\text{s.t.} \quad \alpha_l \geq 0, \quad \sum_{l=1}^M \alpha_l y_l = 0. \quad (16)$$

In the case of linearly nonseparable training data, by introducing slack variables,  $\xi_l$ , the objective problem is given by

$$\min_{\mathbf{w}, b, \xi} \quad \frac{1}{2} \mathbf{w}^T \mathbf{w} + C \sum_{l=1}^M \xi_l \quad (17)$$

$$\text{s.t.} \quad y_l (\mathbf{w}^T K(\mathbf{x}, \mathbf{x}_l) + b) \geq 1 - \xi_l, \quad \xi_l \geq 0,$$

where  $C$  is the regularizing (margin) parameter that determines the trade-off between the maximization of the margin and minimization of the classification error.

Similarly, the corresponding dual problem is expressed as

$$\max \quad L(\alpha) = \sum_{l=1}^M \alpha_l - \frac{1}{2} \|\mathbf{w}\|^2 = \sum_{l=1}^M \alpha_l - \frac{1}{2} \sum_{l=1}^M \sum_{p=1}^M y_l y_p K(\mathbf{x}_l, \mathbf{x}_p)$$

$$\text{s.t.} \quad 0 \leq \alpha_l \leq C, \quad \sum_{l=1}^M \alpha_l y_l = 0. \quad (18)$$

With the OSH found, the decision function can be written as

$$f(\mathbf{x}) = \text{sgn} \left( \sum_{l=1}^M y_l \alpha_l^* K(\mathbf{x}, \mathbf{x}_l) + b^* \right), \quad (19)$$

where  $\alpha_l^*$  and  $b^*$  are the parameters of OSH, respectively.

**2.3. Methodology for Structural Reliability Assessment.** The reliability assessment based on SA and SVM can be implemented as follows.

*Step 1.* Calculate the total sensitivity indices of each input variable in the limit state function,  $f(\mathbf{x})$ , by Monte-Carlo numerical integrations and eliminate those whose total sensitivity indices are less than 0.05.

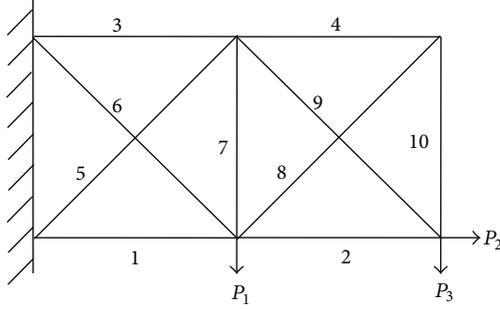


FIGURE 1: 10-bar planar truss.

TABLE 1: Distribution types of random variables.

Variables	$A_i$ (m <sup>2</sup> )	$P_1$ (MN)	$P_2$ (MN)	$P_3$ (MN)
Mean value	0.0001	-160	160	-160
Coefficient of variation	0.15	0.5	0.5	0.5
Distribution types	Normal	Normal	Normal	Normal

*Step 2.* The samples used for the SA in Step 1 are also selected as the train samples for training SVM model. It is noted that the column of train samples is the residual input variable. The number of train samples is defined as  $N_{\text{train}}$ . The set of failure and nonfailure samples are defined as class I and class II, respectively. Use the train samples and associated classes to train the SVM model.

*Step 3.* Produce the test samples of residual variables according to their distributions. The number of test samples is  $N_{\text{test}}$  ( $N_{\text{test}} \gg N_{\text{train}}$ ). The trained SVM model is used to classify the test samples.

*Step 4.* Count the number of samples located in class I. Consequently, the failure probability,  $P_f$ , is

$$P_f = \frac{N_f}{N_{\text{test}}}, \quad (20)$$

where  $N_f$  is the number of test samples located in class I.

### 3. Case Study: 10-Bar Planar Truss

*3.1. General Description.* A numerical 10-bar planar truss has been adopted to validate the proposed reliability assessment method. The young's modulus of each bar is  $E = 2.0 \times 10^8$  KN/m<sup>2</sup>. The sectional area of each bar,  $A_i$  ( $i = 1, \dots, 10$ ), and loads applied to the truss (as shown in Figure 1),  $P_i$  ( $i = 1, \dots, 3$ ), were assumed to be random variables. Their distribution types are listed in Table 1.

The ultimate strength of this material is assumed to be 480 MPa. Consequently, the limit state function,  $f(\mathbf{x})$ , can be expressed as

$$f(\mathbf{x}) = 480 - \max \{ \sigma_1(A_1, \dots, A_{10}, P_1, \dots, P_3), \dots, \sigma_{10}(A_1, \dots, A_{10}, P_1, \dots, P_3) \}. \quad (21)$$

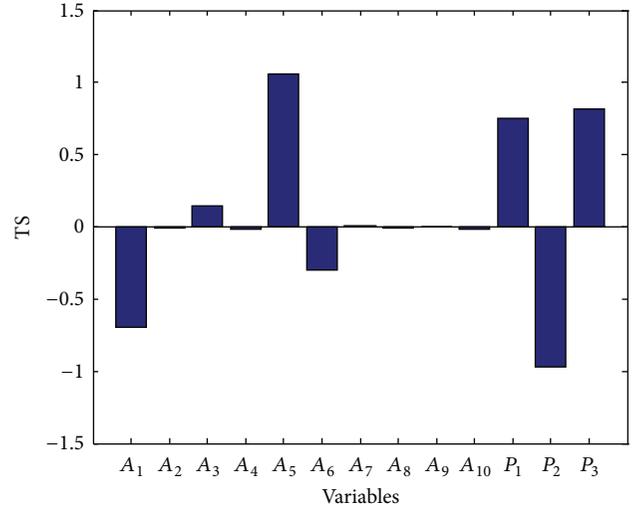


FIGURE 2: Sensitivity indices of each variable.

*3.2. SA of Input Variables.* The Sobol's method was employed to analyze the contribution of each variable on the output variance of limit state function. Firstly, the Latin hypercube sampling technique [23] was used to produce two sets of samples with size of  $50 \times 13$ . Then, the values of  $\hat{f}_0$ ,  $\hat{D}$ , and  $\hat{D}_{\sim i}$  are calculated based on (11)–(13). Finally, the total sensitivity indices of each variable can be obtained by (10). The SA results of input variables are shown in Figure 2. It is found that the sensitivity indices of  $A_2$ ,  $A_4$ ,  $A_7$ ,  $A_8$ ,  $A_9$ , and  $A_{10}$  are less than 0.01, which indicates that these variables have slight contribution on limit state function. Therefore, these variables are rejected as input vectors of SVM model.

*3.3. Support Vector Classifier.* It is noted that a total number of  $14 \times 50 = 700$  samples are calculated in the process of global SA in Section 3.2. These samples are also regarded as training data of SVM but the small-influence variables were rejected as the input vector of SVM model. Therefore, the size of train samples is  $700 \times 7$ . As mentioned in Section 2.3, class I and class II are introduced to describe the failure and nonfailure sample set. The SVM model is trained by use of the train samples and corresponding sample labels. Gaussian radial basis function is ascertained as kernel function of the SVM model. The value of penalty term,  $C$ , and kernel parameter,  $\sigma$ , are determined as  $2 \times 10^6$  and 0.002.

A total number of 20000 test samples are constructed and input into the trained SVM model. The classification results of test samples are shown in Table 2. It is shown in Table 2 that a total number of  $1007 + 399 = 1406$  test samples are classified into class I. Consequently, the corresponding structural failure probability is 7.03%.

### 3.4. Comparisons and Discussions

*3.4.1. Computation Precise.* It is observed in Table 2 that the total CA of test samples is 96.34%, showing an excellent classification capability. However, it is also noted that the CA of

TABLE 2: Classification results.

Method	Samples	Class	Number of samples	Classification number for different classes		CA (%)	Total CA (%)
				I	II		
SA and SVM	Train	I	60	43	17	71.67	96.29
		II	640	9	631	98.59	
	Test	I	1396	1007	389	72.13	96.06
		II	18604	399	18205	97.86	
SVM	Train	I	60	44	16	73.33	96.14
		II	640	11	629	98.28	
	Test	I	1396	902	494	64.61	95.28
		II	18604	451	18153	97.58	

TABLE 3: Failure probabilities evaluated by different methods.

Method	MCSM	RSM	SVM	SA and SVM
Failure probability (%)	6.98	10.29	6.77	7.03
Error (%)	—	47.42	3.01	0.72
Amount of FEA	20000	27	700	700

class I is only 72.13%. The main reasons of this phenomenon can be summed as two aspects: (a) the OSH in SVM model is the approximate limit state function, and the samples nearly the limit state function maybe misclassified. However, it is seen that the number of misclassification samples in class I is 389, almost equal to the number of misclassification samples (399). It indicates that misclassification samples have a slight effect on the assessment results; (b) the sample number of class I is far less than that of class II. When the same number of misclassification samples shows up in classes I and II, the CA of class I drops more rapidly than that of class II.

In order to validate the applicability, other three structure reliability assessment methods (i.e., RSM, MCSM, and SVM) are employed to evaluate the structural failure probability. The failure probabilities evaluated by these four methods are listed in Table 3. Usually, the results from MCSM are regarded as the exact solution. It is found that the result evaluated by the proposed method is closest to that by MCSM, indicating that it is the most precise method among these methods employed except MCSM.

The classification result of SVM model is also listed in Table 2. It is obvious that in total the CA of test samples is 95.28%, slightly less than the proposed method (96.06%). The main reason is that the small-influence input variables affect the shape of OSH, thus causing more distortion of OSH than that of the proposed method.

**3.4.2. Computation Efficiency.** The amount of FEA required by each method is also listed in Table 3. It is found that the least amount of FEA required is RSM, which only needs 27 times. However, its accuracy is the worst. The amount of FEA required by the proposed method and SVM model are far less than the MCSM, while the relative errors are 0.72% and 3.01%, respectively. However, the size of test samples is  $20000 \times 13$ , two times that of the proposed method ( $20000 \times 7$ ). This indicates that the proposed method can largely

reduce the runtime in case of ensuring computation precise in comparison with the SVM model. It can be predicted that the proposed method can largely reduce the data storage memory requirements for the reliability assessment of a more complex structures.

#### 4. Conclusive Remarks

In this study, a novelty reliability assessment method based on SA and SVM has been developed and successfully applied for reliability assessment of a 10-bar planar truss. The results show that the proposed method not only reduces data storage memory requirements with enough computation accuracy, but also has a better assessment capability in comparison with other methods.

The proposed assessment method integrating both SA and SVM is proved to be a successful example. However, it should be noted as well that our success in the proposed method was only achieved through numerical simulations, and more field tests should be done to testify its feasibility and efficiency in practice.

#### Conflict of Interests

We declare that we do not have any commercial or associative interest that represents a conflict of interest in connection with the work submitted.

#### Acknowledgments

The work is supported by the National Natural Science Foundation of China (nos. 51278127 and 50878057), the Ph.D. Programs Foundation of Ministry of Education (no. 20093514110005), and the National 12th Five-Year Research Program of China (no. 2012BAJ14B05), China.

#### References

- [1] S. Mahadevan, *Probability, Reliability, and Statistical Methods in Engineering Design*, Wiley, 2000.
- [2] C. G. Bucher and U. Bourgund, "A fast and efficient response surface approach for structural reliability problems," *Structural Safety*, vol. 7, no. 1, pp. 57–66, 1990.

- [3] R. E. Melchers, *Structural Reliability Analysis and Prediction*, Wiley, 1999.
- [4] K. W. Chau, "Reliability and performance-based design by artificial neural network," *Advances in Engineering Software*, vol. 38, no. 3, pp. 145–149, 2007.
- [5] H. M. Gomes and A. M. Awruch, "Comparison of response surface and neural network with other methods for structural reliability analysis," *Structural Safety*, vol. 26, no. 1, pp. 49–67, 2004.
- [6] A. Basudhar, S. Missoum, and A. Harrison Sanchez, "Limit state function identification using Support Vector Machines for discontinuous responses and disjoint failure domains," *Probabilistic Engineering Mechanics*, vol. 23, no. 1, pp. 1–11, 2008.
- [7] X.-H. Tan, W.-H. Bi, X.-L. Hou, and W. Wang, "Reliability analysis using radial basis function networks and support vector machines," *Computers and Geotechnics*, vol. 38, no. 2, pp. 178–186, 2011.
- [8] J. Shawe-Taylor and N. Cristianini, *Kernel Methods for Pattern Analysis*, Cambridge University Press, 2004.
- [9] H. Aytug and S. Sayin, "Exploring the trade-off between generalization and empirical errors in a one-norm SVM," *European Journal of Operational Research*, vol. 218, no. 3, pp. 667–675, 2012.
- [10] J. E. Hurtado and D. A. Alvarez, "Classification approach for reliability analysis with stochastic finite-element modeling," *Journal of Structural Engineering*, vol. 129, no. 8, pp. 1141–1149, 2003.
- [11] J. E. Hurtado, "An examination of methods for approximating implicit limit state functions from the viewpoint of statistical learning theory," *Structural Safety*, vol. 26, no. 3, pp. 271–293, 2004.
- [12] W.-L. Jin, C.-X. Tang, and J. Chen, "SVM based on response surface method for structural reliability analysis," *Chinese Journal of Computational Mechanics*, vol. 24, no. 6, pp. 713–718, 2007 (Chinese).
- [13] Z. Guo and G. Bai, "Application of least squares support vector machine for regression to reliability analysis," *Chinese Journal of Aeronautics*, vol. 22, no. 2, pp. 160–166, 2009.
- [14] E. Plischke, "An effective algorithm for computing global sensitivity indices (EASI)," *Reliability Engineering and System Safety*, vol. 95, no. 4, pp. 354–360, 2010.
- [15] I. M. Sobol', "Sensitivity analysis for nonlinear mathematical models," *Mathematical Modelling and Computational Experiment*, vol. 1, pp. 407–414, 1993.
- [16] I. M. Sobol', "Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates," *Mathematics and Computers in Simulation*, vol. 55, no. 1–3, pp. 271–280, 2001.
- [17] I. M. Sobol' and Y. L. Levitan, "On the use of variance reducing multipliers in Monte Carlo computations of a global sensitivity index," *Computer Physics Communications*, vol. 117, no. 1, pp. 52–61, 1999.
- [18] I. M. Sobol', "Theorems and examples on high dimensional model representation," *Reliability Engineering and System Safety*, vol. 79, no. 2, pp. 187–193, 2003.
- [19] I. M. Sobol', S. Tarantola, D. Gatelli, S. S. Kucherenko, and W. Mauntz, "Estimating the approximation error when fixing unessential factors in global sensitivity analysis," *Reliability Engineering and System Safety*, vol. 92, no. 7, pp. 957–960, 2007.
- [20] I. M. Sobol' and S. Kucherenko, "A new derivative based importance criterion for groups of variables and its link with the global sensitivity indices," *Computer Physics Communications*, vol. 181, no. 7, pp. 1212–1217, 2010.
- [21] J. W. Hall, S. Tarantola, P. D. Bates, and M. S. Horritt, "Distributed sensitivity analysis of flood inundation model calibration," *Journal of Hydraulic Engineering*, vol. 131, no. 2, pp. 117–126, 2005.
- [22] R. I. Cukier, H. B. Levine, and K. E. Shuler, "Nonlinear sensitivity analysis of multiparameter model systems," *Journal of Computational Physics*, vol. 26, no. 1, pp. 1–42, 1978.
- [23] J. C. Helton and F. J. Davis, "Latin hypercube sampling and the propagation of uncertainty in analyses of complex systems," *Reliability Engineering and System Safety*, vol. 81, no. 1, pp. 23–69, 2003.

## Research Article

# An Algorithm for Mining of Association Rules for the Information Communication Network Alarms Based on Swarm Intelligence

Yang Wang,<sup>1,2</sup> Guocai Li,<sup>1,2</sup> Yakun Xu,<sup>1,2</sup> and Jie Hu<sup>1,2</sup>

<sup>1</sup> MicroNano System Research Center, College of Information Engineering, Taiyuan University of Technology, Taiyuan, Shanxi 030024, China

<sup>2</sup> Key Lab of Advanced Transducers and Intelligent Control System of the Ministry of Education, Taiyuan University of Technology, Taiyuan, Shanxi 030024, China

Correspondence should be addressed to Jie Hu; [hujie0351@gmail.com](mailto:hujie0351@gmail.com)

Received 5 August 2013; Accepted 17 November 2013; Published 19 January 2014

Academic Editor: Orwa Jaber Housheya

Copyright © 2014 Yang Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Due to the centralized management of information communication network, the network operator have to face these pressures, which come from the increasing network alarms and maintenance efficiency. The effective analysis on mining of the network alarm association rules is achieved by incorporating classic data association mining algorithm and swarm intelligence optimization algorithm. From the related concept of the information communication network, the paper analyzes the data characteristics and association logic of the network alarms. Besides, the alarm data are preprocessed and the main standardization information fields are screened. The APPSO algorithm is proposed on the basis of combining the evaluation method for support and confidence coefficient in the Apriori (AP) algorithm as well as the particle swarm optimization (PSO) algorithm. By establishing a sparse linked list, the algorithm is able to calculate the particle support thus further improving the performance of the APPSO algorithm. Based on the test for the network alarm data, it is discovered that rational setting of the particle swarm scale and number of iterations of the APPSO algorithm can be used to mine the vast majority and even all of the association rules and the mining efficiency is significantly improved, compared with Apriori algorithm.

## 1. Introduction

The operation and maintenance management of information communication network mainly refers to timely discovery, locating and handling of any network fault to ensure smooth and efficient operation as well as guarantee in major emergencies pertinent to network operation, complaints about network quality from customers, assessment and analysis of network quality, prediction of planning, construction, and so forth. The time consumed during fault location and judgment in the application layer of a large-scale network accounts for 93% of its total time for failure of recovery [1]. The huge network structure and multifunctional device types also bring about large amounts of alarm data due to such characteristics of the information communication network

as topological structure densification, network device micro-miniaturization, communication board precision, and so forth. Therefore, the foundation of the network operation and maintenance is the effective management of the network alarms.

As an important supporting means for network operation and maintenance management, network management system directly influences the quality of service which the information communication network provides to its customers [2]. The network management system is developing toward integrated service network management update from independent device network management, manufacturer device network management, and integrated professional network management. The centralized monitoring management function of the professional information communication network

operation management will make problems exhibit a sharp full data increasing, including network faults, device alarms, and customer complaints.

As the information communication system consists of various medium interlinked network devices and operating systems implicit and complex-correlated logic is ubiquitous among network elements; that is, a certain fault point may trigger numerous alarms in the whole network. The sudden intensive alarms not only consume the resources of the network management system but also obscure the position of the network fault source points thus severely impeding trouble shooting by the network operation and maintenance personnel. Several alarms are incorporated into a single alarm or source alarm with a large amount of information by such links as paraphrasing and explaining, eliminating and filtering, information integration, and correlating and transforming, and so forth. It aims at assisting the operation and maintenance personnel to analyse fault messages and locate faults quickly, that is, mining analysis on alarm association rules.

Mining of alarm association rules refers to a process of analysis on the association between the attributive characteristic logic of the alarms within devices and the topological hierarchy of network devices. It aims at achieving clear critical alarms, accurate fault location and troubleshooting, and intelligent fault prediction and evaluation. The mining of alarm association rules can be divided into three levels: analysis on alarm association in the device within the profession, analysis on topological alarm association of the network device within the profession, and analysis on inter-professional topological alarm association of the network device but their core is mining algorithm for association rules [3].

The centralized management of information communication network brings about large amounts of alarm data. A rapid mining analysis on the network alarm association rules is achieved by the classic Apriori association mining algorithm and PSO algorithm under the context of big data. The alarm association relationship can be used to add and merge the fault alarms, maintain the work order, improve the centralized monitoring efficiency, and reduce the cost of network maintenance.

The Apriori is an association rules mining algorithm based on characteristics of frequent item sets (priori knowledge) whose core concept is a layer-wise iterative search of the theory of frequent item sets. However, the Apriori algorithm thought also presents some inevitable problems. For instance, frequent repeated scans of the information in the sample database lead to a heavy load on the system I/O; large item sets lead to a sharp increase in the number of the candidate frequent item sets and a significant increase in operation time, and so forth.

Swarm intelligence refers to the macroscopic intelligent group behavior showed by various types of organism individuals in the nature during survival, collaboration, and evolution. Application research is conducted for the swarm intelligence algorithm in optimization solutions of engineering problems such as economic analysis and forecast, structural damage positioning and inspecting, command and

dispatch of communication and transportation, evacuation route planning, target identifying and tracking, factory site selection and evaluation, communication network planning, and route plan preparation [4]. The swarm intelligence algorithm has such advantages as distributed control, indirect information transfer, and simple individuals and swarm intelligence. As a classic swarm intelligence algorithm, the particle swarm optimization also has the above characteristics.

Centralized management of the alarms in the information communication network is an important part of operation maintenance of the information communication network. The alarm correlation directly influences the quantity and quality of the alarm work orders. An analysis on the large amounts of alarm data through an efficient algorithm becomes the critical technical means. The APPSO discussed in the paper incorporates the Apriori algorithm and swarm optimization algorithms and applies swarm optimization algorithms in the information communication field.

Section 2 in the paper elaborates such basic concepts of faults in the information communication network, network alarms, alarm standardization, and so forth; Section 3 discusses the data characteristics of network alarms, and the alarm correlation logical relationships within and between network devices; Section 4 describes achieving quality improvement of the data source of the network alarms by pre-processing of the network alarm data; Section 5.1 presents the concepts of support and confidence coefficient and mining analysis process in Apriori algorithm in combination with examples; Section 5.2 describes the swarm intelligence model and basic flow of the PSO algorithm; Section 5.3 discusses the creation of APPSO association rule mining algorithm, which deducts on the basis of the Apriori and PSO algorithm characteristics. Besides, combining with the characteristics of the network alarm data, the section puts forward the improvement of the performance of the APPSO association rule mining algorithm by sequencing code, sliding window, sparse linked list, and nature of the Apriori algorithm. It conducts a performance test for the algorithm through the alarm data in the information communication network from different angles. At the end of the section, an index evaluation of the alarm association rate is put forward, which is used for application of the alarm correlation relationship derived from the APPSO algorithm mining into the actual network.

## 2. Concepts Pertinent to Alarms in the Information Communication Network

Concepts pertinent to the data analysis on the alarms in the information communication network are defined as follows [5, 6].

*Definition 1.* A network fault refers to an event where the information communication network is not able to operate normally and efficiently due to some reasons and even no service can be provided. The reasons causing network faults can be divided into network device faults, communication link abnormality, inappropriate operation and maintenance,

energy power and room environment abnormality, and network system faults (affecting monitoring instead of the communication service).

*Definition 2.* A network alarm is a message triggered during abnormal operation of communication device and each alarm message represents its unique running status. No uniform standard specification is applicable to the network devices in the whole industry due to the difference in mechanism and connotation of the alarm messages of devices of different types from various manufacturers. However, the standardization can be achieved by specific standardized fields.

*Definition 3.* Alarm standardization redefines the level, classification, influence, and so forth, of the full professional alarms, which achieving the target on achieve mapping definition, normative classification, and centralized management of professional alarms of different manufacturers.

*Definition 4.* The alarm standardization fields include profession, manufacturer, device type, alarm title, auxiliary fields of alarm explanation, manufacturer alarm level, applicable manufacturer version number, network management alarm level, network management alarm ID, alarm explanation, alarm class, alarm logic class, alarm logic subclass, effect of such an event on the device, effect of such an event on the service, and standard name of the alarm.

*Definition 5.* The alarm standardization fields of the network management system refer to the other alarm standardization fields of the network management system excluding the alarm standardization fields, for example, city/county/district, network element name, number of network element board card, local port information of the alarm, remote port information of the alarm, occurrence time of the network element alarm, discovery time of the network management alarm, elimination time of the alarm, and so forth.

### 3. Data Characteristics and Association Logic of Network Alarms

The information communication network has such characteristics as complex, hierarchical, and full end-to-end networking. These network elements have certain physical and logical association, and the independent network element failure will result in “click alarm, multclick dissemination” effect on related network element. However, there is association of occurrence time and logical name between these alarms. Thus, association, classification, and combination of such alarms can substantially improve the efficiency of centralized monitoring [7].

*3.1. Data Characteristics of Network Alarms.* Information communication network alarm is characterized by huge data volume, alarm fluctuation, network communication effect, accumulative and lagging effects and redundancy of fault messages, and so forth. The analysis of these characteristics

will contribute to mining analysis on rules of association among alarms.

(1) *Huge Data Volume.* The number of alarms and faults in the current network is huge due to such characteristics as diversification of types of information communication network services, network scale expansion, topological structure tightness and centralization of network monitoring, and so forth.

(2) *Alarm Fluctuation.* From the perspective of monitoring management, the equipment failure alarms have certain unpredictability. The crash of critical equipment will cause the whole network paralysis leading to a sharply increasing number of alarms inevitably. Similarly, the alarms can be eliminated if the failures are maintained and handled timely. For instance, the block of central transmission lines will affect local lines, lines across cities, and relative network equipment; thus, all relevant equipment exhibits alarm conditions. If the central lines are dealt with appropriately, the alarm will be removed rapidly.

(3) *Network Communication Effect.* The alarm does not spread through some concrete networks but relies on the independent “management network” [8]. Take SDH network alarm for example, LAN regenerator section LOS alarm → multiplex section MIS → AIS alarm → remote device MS-FERF alarm connected to local devices and AU-AIS alarm → local HO-VC HP-AIS alarm → local TU-AIS alarm and HP-FERF/RDI alarm.

(4) *Accumulative and Lagging Effect.* The abnormality of some network equipment would degrade the relative network quality. If this condition has accumulated to an extent that exceeds the limits, the connected network equipment would alarm. Besides, these features may be caused by clock synchronous exception among communication equipment, NM for manufacturer’s equipment and NM for multidisciplinary or abnormal network management data.

(5) *Redundancy of Fault Messages.* Fault points on single panel would cause the associated devices parts to alarm; and the failure of network convergence nodes can trigger a large-scale network alarm. For example, the failure of MSC server (mobile switching center) will lead some devices to stay in an alarm state, such as, MGW (media gateway), BSC (base station controller), and RNC (radio network controller). And this phenomenon will lead to a sudden “alarm storm.”

(6) *Abundant Property Field.* Each alarm corresponds to some recognized information combination. Different property fields reveal certain relevant logic.

(7) *Abnormal Alarm.* It can be divided into waste alarm, ultrashort alarm, and overlength alarm. The waste alarm is not caused by the filter clear of network access test and device data in time. The ultrashort alarm points the alarm lasts for less than one minute. And the overlength alarm refers to the alarms which are not removed after a long time.

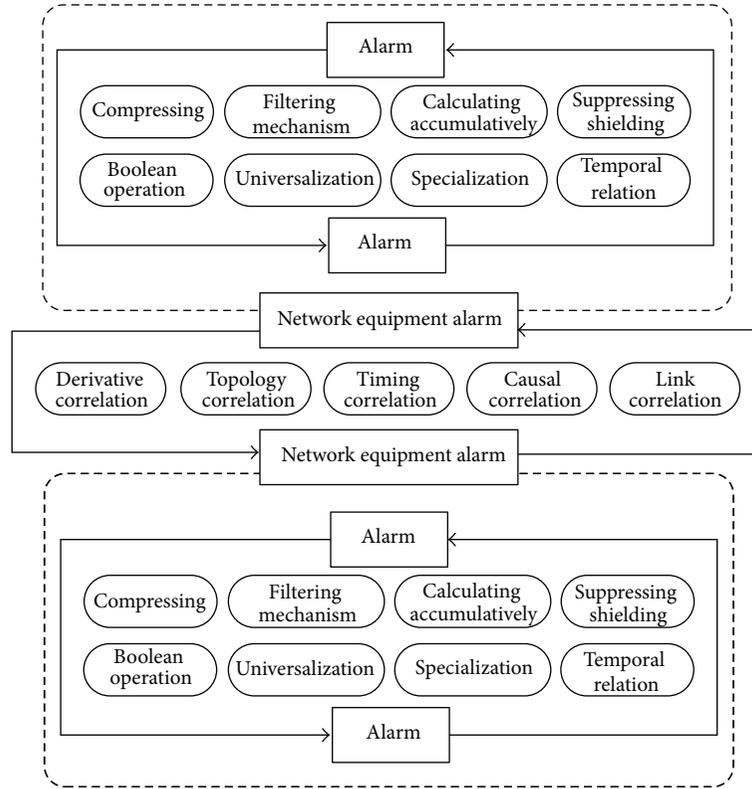


FIGURE 1: Association logic of network alarms.

3.2. *Association Logic of Network Alarms.* The network association logic can be divided into two levels, that is, alarm association logic within network device and alarm association logic among network device as shown in Figure 1.

The alarm logical association on the network equipment itself is as follows [9]: (1) alarm compressing: taking the simultaneous multialarm which has the same attributes (adjacent cells, same network element or light path, etc.) into an alarm; (2) filtering mechanism: alarm which does not conform to the attribute association will be deleted; (3) calculating accumulatively: a number of concurrent alarms will be converted to an alarm with new name; (4) suppressing shielding: low priority alarms will be suppressed when they are of high priority to be generated; (5) boolean operation: making a group of alarms in conformity with some rules of Boolean operation into an alarm; (6) generalization: network element is to be a more general alarm; (7) specialization: the more detailed alarm information will replace network element alarms; (8) temporal relation: the different alarms are to be generated as per certain time sequence.

Alarm association among groups of network equipment is as follows: (1) derivative association: the network equipment alarms are divided into root alarm and derivative alarm; (2) topological association: the network equipment alarm contains home terminal alarm and opposite end; (3) timing association: the same fault point generates alarms with the same time trigger characteristic; (4) causal association: Occurrence of Alarm A causes Alarm B, that is, element management system has been out of management as a result of optical cable break; (5) link association: convergence line

fault will trigger the network equipment alarm on the entire path and send unification orders.

#### 4. Preprocessing of Network Alarm Data

The transmission network device alarm data is used as the analytical data for association rules for the information communication network alarms and the link of data preprocessing is as follows (Figure 2).

(1) *Data Extraction.* All transmission alarms within a specific time interval are extracted through the network management system (including engineering cutover and device alarms arising from network adjustment) and the data fields extracted include alarm standardization field and network system alarm standardization field.

(2) *Data Cleaning.* Special data affecting the algorithm analysis quality is cleaned from the alarm data extracted and such data includes

- ① abnormal data: junk alarm, ultrashort alarm, ultralong alarm, and abnormal and special alarm data,
- ② incomplete data: alarm data with a null alarm determinant attribute field,
- ③ erroneous data: alarm data with a large difference between the time field of the network management alarm and the time field of the device alarm due to time synchronization abnormality,

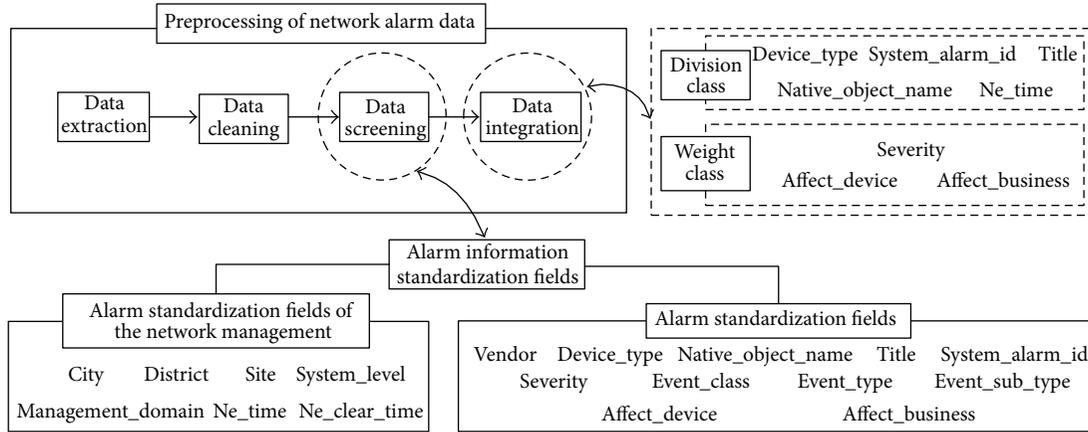


FIGURE 2: Preprocessing of network alarm data.

④ duplicated data: duplicated alarm data due to merging or removing flashes.

(3) *Data Screening*. ① Interference data: screen and reject the interference alarm data, for example, uncorrelated alarms (alarms such as access control enabling and mismatching of the main and standby sing board versions) in a number of signal alarms (alarms such as signal degradation indication and output signal loss) are rejected. During screening, the duplicated alarms should not be deleted blindly and they should be analyzed and discriminated based on the actual fault conditions considering that the duplicated alarms may be caused by different faults during different periods [10].

② Alarm information standardization field: main information fields are screened from the standardization fields of the network management alarms and alarm standardization fields for subsequent mining of association rules. These information fields are set as two classes: division class and weight class. The alarm information fields of division class are mainly used to describe attribution relation and attribute parameters of alarms. The alarm information fields of weight class are mainly used to describe importance difference and influence and assign differentiated weight to the data of the association rule mining algorithm.

(4) *Data Integration*. The alarm processed in the above link and its corresponding information standardization fields are resorted out eventually and generate network alarm data sources with high information amount.

### 5. Mining Algorithm for Association Rules for the Network Alarm Data

The Apriori algorithm has been widely used by researchers as a classic mining algorithm for association rules. While the swarm intelligence algorithm has been studied deeply and applied in various fields due to its characteristics such as distributed control, low communication overhead, simple behavior rule, and strong self-organization. The APPSO

algorithm is exactly an efficient algorithm for it incorporates the above two algorithm thoughts and combines with the data characteristics of the alarms in the information communication network.

5.1. *Example Analysis for Apriori Algorithm*. On the ICDM (IEEE International Conference on Data Mining) held in December 2006, the top ten classical algorithms were selected from the 18 candidate algorithms after three links of nomination, review, and voting, that is, C4.5 (classification), K-Means (statistical learning), SVM (statistical learning), Apriori (association analysis), EM (statistical learning), PageRank (link mining), AdaBoost (bagging and boosting), kNN (classification), Naive Bayes (classification), and CART (classification). The Apriori algorithm formulated by Wu and Vipin in 2009 ranks fourth among the ten top classical algorithms for data mining, which also sufficiently shows its importance in data mining algorithm [11].

The association rules mining algorithm exactly obtains the association relationship among terms from data sets through mathematical logic. The market basket analysis sufficiently embodies the industrial application value of the association rules mining algorithm. The Apriori is an association rules mining algorithm based on characteristics of frequent item sets (a priori knowledge) whose core concept is a layer-wise iterative search of the theory of frequent item sets.

In combination with the examples of fault alarms of the information communication network, the application of concept and flow of the Apriori algorithm are discussed as follows.

#### 5.1.1. Concept of the Apriori Algorithm

- (1) All item sets: all alarm item sets of the examples, that is, Alarm1–Alarm5,
- (2) item set: concurrent item combination, for example, {Alarm1, Alarm2}, {Alarm2, Alarm3, Alarm4},

- (3) support: describes universality and frequency of association rules and the association rule of high support reflects that it may be applicable to most events of the data sets,
- (4) support count: the number of alarm affairs contained in a group of item sets,
- (5) confidence: describes reliability and accuracy of the association rules, that is, probability of Alarm2 occurrence on the premise of Alarm1 occurrence (conditional probability).

As for the mining association rules of the Apriori algorithm, high support and low confidence of the association rule indicate the reliability of the association rule is poor; low support and high confidence of the association rule indicate the applicability of the association rule is poor. Minimum support count and minimum confidence are set manually by users. An association rule is deemed to be concerned if it satisfies both parameters above [12]. The matching relation between the support and the confidence should be set rationally in combination with the value demand for industrial rules in practical application.

The generation process of association rules is also the process where joining, pruning, and enumerating are performed through support and confidence. The association rules are not able to be applied directly through the algorithm; besides, the application value requires analyzing and screening by experts.

*5.1.2. Flow of the Apriori Algorithm.* The flow the Apriori algorithm can be reduced to the following steps [13]: (1) analysing the frequent item sets, that is, obtaining all item sets no less than the preset minimum support count from the iteration of the full database (joining, pruning, and enumerating); (2) obtaining the strong association rules, that is, extracting the association rules from the frequent item sets based on the minimum support and minimum confidence. In combination with instances, the analysis and explanation are presented in Table 1.

Table 1 shows the corresponding alarm items generated on the network device when the information network fails. The network fault events are successively defined as Fault 1–Fault 5. The alarm item class corresponding to each fault is defined as Alarm1–Alarm5 (reduced to A1–A5). The network faults arising from different reasons will generate different combinations of alarm item classes (Table 1).

(1) All alarm item sets are scanned and the support of each alarm item is calculated in Table 2.

(2) The minimum support count is 2 and the candidate item set C1 will form after screening (eliminating A5) of the alarm item combinations in L1 (see Table 3).

(3) All alarm item sets are scanned again to form the support calculation L2 based on the candidate item set C1 (see Table 4).

(4) The minimum support count is 2 and the candidate item set C2 will form after screening (eliminating {A1, A4} and {A3, A4}) of the alarm item combinations in L2 (see Table 5).

TABLE 1: Fault alarms of information communication network.

Network fault event (fault ID)	Alarm item sets (alarm items)
Fault 1	A1, A2, A3, A5
Fault 2	A2, A4
Fault 3	A2, A3
Fault 4	A1, A2, A3, A4
Fault 5	A1, A3

TABLE 2: Calculation of support of all alarm item sets L1.

Alarm item	Support count
A1	3
A2	4
A3	4
A4	2
A5	1

TABLE 3: Support of alarm item sets C1.

Alarm item	Support count
A1	3
A2	4
A3	4
A4	2

TABLE 4: Support of alarm item sets L2.

Alarm item	Support count
{A1, A2}	2
{A1, A3}	3
{A1, A4}	1
{A2, A3}	3
{A2, A4}	2
{A3, A4}	1

TABLE 5: Support of Alarm Item Sets C2.

Alarm item	Support count
{A1, A2}	2
{A1, A3}	3
{A2, A3}	3
{A2, A4}	2

TABLE 6: Support of alarm item sets L3.

Alarm item	Support count
{A1, A2, A3}	2
{A1, A2, A4}*	1
{A1, A3, A4}*	1
{A2, A3, A4}*	1

(5) All alarm item sets are scanned again to form the support calculation L3 based on the candidate item set C2 (see Table 6). Based on the nature of the Apriori algorithm (all subsets of the item sets are frequent necessarily), {A1, A4}

and  $\{A3, A4\}$  are not frequent item sets. Thus,  $\{A1, A2, A4\}^*$ ,  $\{A1, A3, A4\}^*$ , and  $\{A2, A3, A4\}^*$  in Table 6 are not frequent item sets and can be excluded directly.

(6) The minimum support count is 2 and the final item set C3 will form after screening of the alarm item combinations in L2 (see Table 7).

The nonvoid proper subsets of  $\{A1, A2, A3\}$  include  $\{A1, A2\}$ ,  $\{A1, A3\}$ ,  $\{A2, A3\}$ ,  $\{A1\}$ ,  $\{A2\}$  and  $\{A3\}$ , and it can be inferred that the confidence coefficients are as presented in Table 8.

They meet the confidence coefficient confidence = 60% and the association rules are obtained  $\{A1, A2\} \rightarrow A3$ ,  $\{A1, A3\} \rightarrow A2$ ,  $\{A2, A3\} \rightarrow A1$ ,  $A1 \rightarrow \{A2, A3\}$ ; that is, Alarm3 will necessarily appear when Alarm1 and Alarm2 occur concurrently; the probability of concurrent occurrence of Alarm2 and Alarm3 is 67% when Alarm1 occurs; the rules for others are similar.

Based on the thinking of the Apriori algorithm flow above, the characteristics are as follows.

(1) Advantages: the algorithmic logic is clear without any complex mathematical derivation process with the dual parameter values of the support and confidence coefficient as the interest indicator for weighing the association rules.

(2) Disadvantages: frequent repeated scans of the information in the sample database lead to a heavy load on the system I/O; the number of the candidate frequent item sets increases sharply and the operation time increases significantly when the item sets are large; the attribute difference and importance of the set elements is ignored and high-value information is lost when the support and confidence coefficient serve as the sole criterion for weighing the item sets; the single-dimensional Boolean type association rules mining mode is used and multidimensional, multilevel, and numeric type association rules need to be improved.

In response to disadvantages of the Apriori algorithm, researchers compress the database samples by random sampling, formulate hash functions to the size of the candidate item set, reduce the number of scanning of the database by the method of dynamic item set counting, quickly establish frequent item sets utilizing the relation of "local-overall," optimize the event database to reduce the quantity of the item sets in combination with the nature of the Apriori algorithm, use parallel computation, and so forth [14–16].

Based on the Apriori algorithm thought, Han et al., a professor from Simon Fraser University, adopted a partition search method combining expanded prefix tree data structure and branch-like local growth, that is, FP-growth (frequent pattern-growth) algorithm in 2000 [17], which avoids the problem of repeating an ergodic database in the Apriori algorithm and substantially improves the mining efficiency of association rules.

**5.2. Particle Swarm Intelligence Algorithm.** The adaptivity and high-efficiency characteristics of group system consisting of the natural ecosystem and various kinds of organisms in response to complex problems (e.g., community cooperation, biological evolution, immune system, nerve conduction, etc.) provide new research directions and application schemes

TABLE 7: Final item set of alarm item sets C3.

Alarm item	Support count
$\{A1, A2, A3\}$	2

for complex scientific problems, for example, ant colony algorithm, bat algorithm, bee algorithm, firefly algorithm, cuckoo search algorithm, particle swarm optimization algorithm, and so forth [18]. In 1987, the zoologist Reynolds simulated the process of aggregating and flying of bird flock self-organization by establishing flight rules for individuals of the bird flock, that is, collision avoidance, velocity matching, and flock centering [19]. In 1995, Kennedy and Eberhart analysed the process of aggregating, scattering, and migrating of birds; that is, when a bird flock searches for specific food in a certain unknown area at random, all individuals of the bird flock do not know their locations but they know the distance between their locations and the food. The simplest and efficient strategy is to search for the peripheral region of the bird closest to the food [20]. The whole foraging process achieved information sharing and competitive collaboration among individuals of the low-intelligence bird flock. In addition, the process embodies the value of the group intelligence evolving from unordered to ordered in obtaining the optimum solution. Kennedy considered the individuals of the birds as single particles and proposed the particle swarm optimization (PSO). The whole process follows the principles of environmental stimulus evaluation, adjacent individuals comparison, and learning adjacent advanced individual [21].

The PSO algorithm first initializes the particle swarm; that is, random location and velocity are assigned to the particle swarm in the feasible solution space. Each particle is a feasible solution in the optimization problem. A fitness value is determined by an optimization function; then, each particle will move in the solution space and the particle velocity will determine its motion direction and distance. Usually, particles approximate the current optimal particle until the optimal solution by means of iteration and each particle will approximate two optimal solutions during iteration, that is, particle optimum solution (POS) and global optimum solution (GOS).

**5.2.1. Fundamental Principles of PSO.** Assume a  $d$ -dimensional target search space, there is a group of particle swarms consisting of  $m$  particles with potential problem solution  $S$ ,  $s = \{\vec{x}_1, \vec{x}_2, \dots, \vec{x}_m\}$ , among which  $\vec{X}_i = (x_{i1}, x_{i2}, \dots, x_{id})$ ,  $i = 1, 2, \dots, m$  indicates a vector point of  $i$ th in the  $d$ -dimensional solving space;  $\vec{x}_i$  is substituted into the objective function pertinent to solving problem and the matched fitness value can be obtained.  $\vec{P}_i = (p_{i1}, p_{i2}, \dots, p_{id})$  is used and  $i = 1, 2, \dots, m$  indicates the optimum value point of the  $i$ th particle obtained by self-search (the optimum value means that its corresponding fitness value is the minimum); in the particle swarm  $S$ , there is an overall optimum particle, which is calculated as  $\vec{G}_i = (g_{i1}, g_{i2}, \dots, g_{id})$ ,  $i = 1, 2, \dots, m$ ; each particle also has a velocity variable

TABLE 8: Calculation of confidence coefficient of C3 alarm item set.

Logical relationship among alarms	Analytical calculation of confidence coefficient
$\{A1, A2\} \rightarrow A3$	Confidence = $2/2 = 100\%$
$\{A1, A3\} \rightarrow A2$	Confidence = $2/3 \approx 67\%$
$\{A2, A3\} \rightarrow A1$	Confidence = $2/3 \approx 67\%$
$A1 \rightarrow \{A2, A3\}$	Confidence = $2/3 \approx 67\%$
$A2 \rightarrow \{A1, A3\}$	Confidence = $2/4 = 50\%$
$A3 \rightarrow \{A1, A2\}$	Confidence = $2/4 = 50\%$

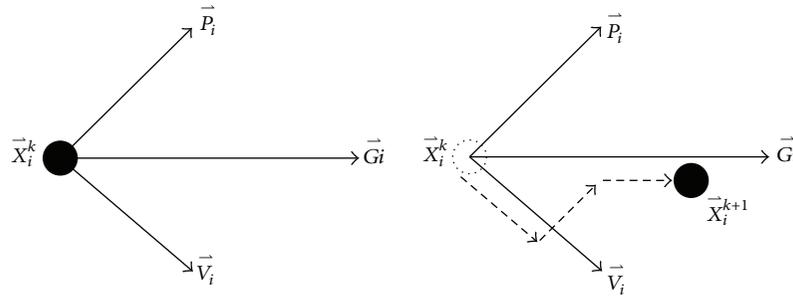


FIGURE 3: Particle migration of PSO.

$\vec{V}_i = (v_{i1}, v_{i2}, \dots, v_{id})$ ,  $i = 1, 2, \dots, m$  indicating the velocity of the  $i$ th particle.

In the PSO algorithm, the following formulae are used for recursive calculation of particle movement:

$$\vec{V}_i^{k+1} = \vec{V}_i^k + c_1 * r_1 * \left( \vec{P}_i^k - \vec{X}_i^k \right) + c_2 * r_2 * \left( \vec{G}_i^k - \vec{X}_i^k \right), \quad (1a)$$

$$\vec{X}_i^{k+1} = \vec{X}_i^k + \vec{V}_i^{k+1}, \quad (1b)$$

where the particle number is  $i = 1, 2, \dots, m$ ;  $k$  is the number of iterations; learning factors  $c_1$  and  $c_2$  are positive constants to which 2 is usually assigned;  $r_1$  and  $r_2$  are random numbers distributed between  $[0, 1]$ . In order to maintain the values of  $\vec{V}_i^k$  and  $\vec{X}_i^k$  within a reasonable regional range  $\vec{V}_{\max}$  and  $\vec{X}_{\max}$  should be set rationally.

Formula (1a) encompasses three facets of information when calculating the new velocity  $\vec{V}_i^{k+1}$  of the particle  $i$ : firstly, velocity  $\vec{V}_i^k$  is the speed of the particle  $i$  at the previous moment, secondly, information on distance between the current position of the particle  $i$  and the optimum position of the individual particle, and thirdly, the information on the current position of the particle  $i$  and the optimum position of the overall particle swarm. Formula (1a) is deployed to calculate the new position coordinates of particles. Formula (1a) and formula (1b) jointly determine the next motion position of the particle  $i$ . Taking a two-dimensional space as an example, Figure 3 describes the process where a particle moves from its initial position to its new position based on formula (1a) and formula (1b).

From the social dynamics, an analysis is conducted: the first part of formula (1a) is the memory term reflecting the velocity vector of particle in the previous step; the

second part is self-recognition term, a vector pointing to the optimum point of the particle from the current point, reflecting self-learning judgment of the particle under the effect of ambient particle swarm; the third part is the group-recognition term, a vector pointing to the optimum point of the overall particle swarm from the current point, reflecting experience sharing and collaboration among particles. The process reflects the basic learning development rules for biotic communities in the nature, that is, the process where companion knowledge learning and self-cognitive decision-making are integrating under constant action of external environmental information.

**5.3. Optimization Algorithm for Mining of Particle Swarm Association Rules.** Based on an analysis of the flow for the Apriori algorithm and particle swarm optimization, it has been discovered that the process of searching for the frequent items in the Apriori algorithm is actually a global search process while the particle swarm optimization is an algorithm finding the optimal global solution with excellent optimal performance. Therefore, the global optimum characteristic of the Apriori algorithm and the high efficiency of seeking the global optimal solution of the particle swarm optimization are needed for combing to achieve the optimization algorithm for association rules mining-APPSO algorithm.

**5.3.1. Basic Flow of the APPSO Algorithm.** The Apriori algorithm includes two stages and its overall performance is primarily determined by the first link, which aims at finding all frequent item sets meeting the minimum support in the database; the second link refers to finding the association rules meeting the minimum confidence coefficient from the frequent item sets.

Create three particle swarms in APPSO algorithm (see Figure 4), that is, the sample particle swarm, the candidate

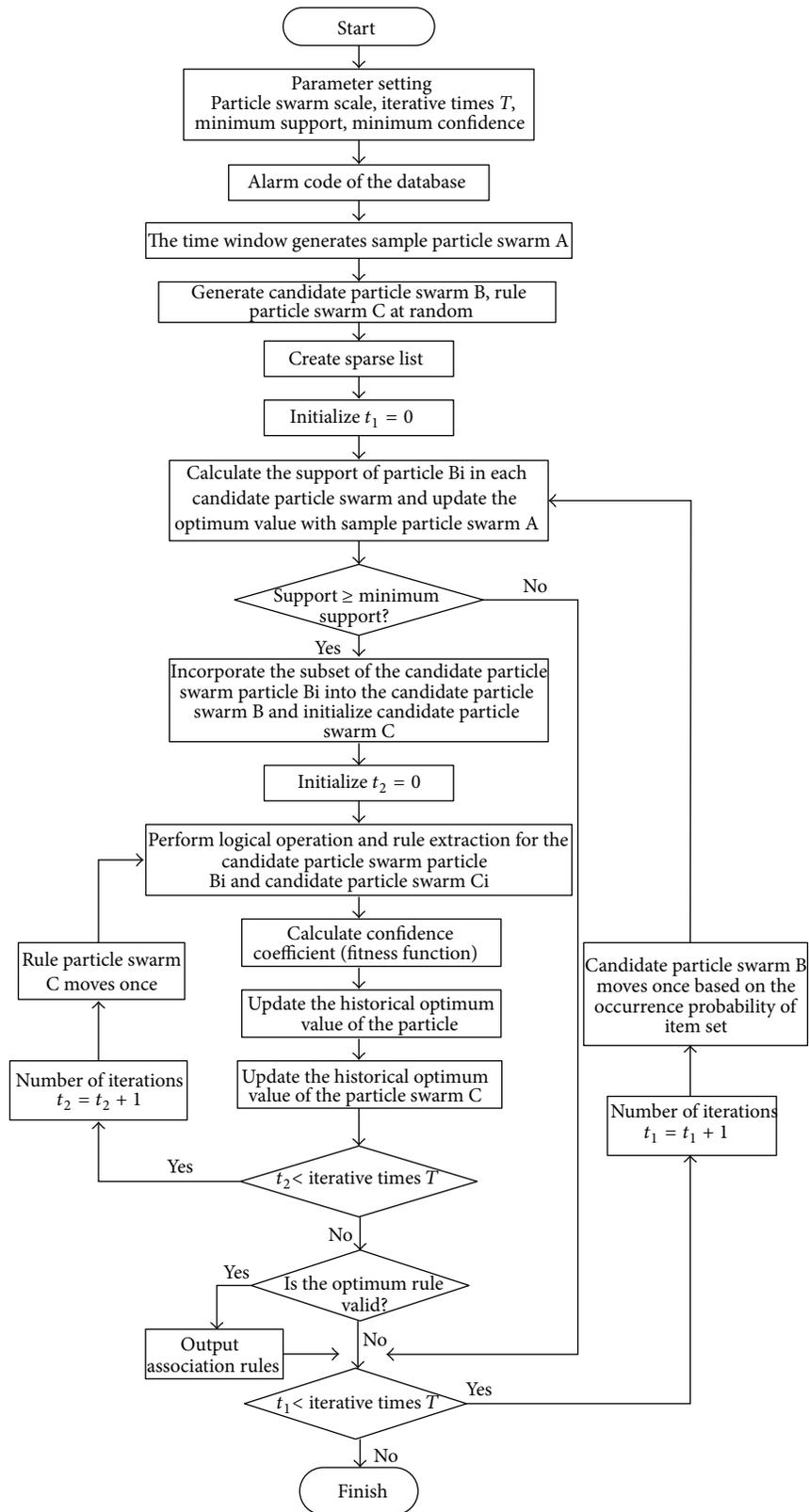


FIGURE 4: Basic flow of the APPSO algorithm.

particle swarm, and the rule particle swarm. The sample particle swarms are entity particle swarms; taking four-dimensional alarm data as an example, the sample particles are (A1, A3, A2), (A1, A2, A4); the candidate particle swarm and the rule particle swarm are logical particle swarms, for example, (1110) and (1101). The eligibility of the particles in the candidate particle swarm for candidate particles is determined by calculating and determining whether the particles in the sample particle swarm satisfy the minimum support. The particles in the candidate particle swarm and the rule particle swarm are judged logically to generate preliminary association rule. The association rules will be output if each preliminary association rule satisfies the minimum confidence; otherwise they will be discarded. The creating process is as follows.

(i) Sample particle swarm: the alarm data source is partitioned to create sample particle swarm A (SPS-A for short) by sliding the time window. For instance, after the number  $N$  time window capturing the natural time, the alarm sequence is shown in A1, A3, and A4; namely, the particle is A1, A3, and A4.

(ii) Candidate particle swarm: B particle swarm is created randomly in the APPSO algorithm (corresponding to the first link in the Apriori algorithm) such that each particle of the candidate particle swarm represents a certain candidate item set and all candidate particles of the whole candidate particle swarm represent a collection of all existing different candidate item sets. The support of the item set represented by each candidate particle is calculated to judge whether it meets the minimum support count value (calculation method, see Section 5.1.2). Such a particle swarm is referred to as candidate particle swarm B (Particle swarm CPS-B).

It is assumed that there are 4 types of alarms in the alarm database and they are Alarm A1, A2, A3, and A4, respectively. Each alarm is expressed with 0 or 1. 0 indicates that the alarm is not in the candidate particle currently while 1 indicates that the alarm is in the candidate particle currently. It is assumed that the value of a candidate particle is 1100; that is, Alarm A3 and Alarm A4 are not in the candidate particle and the particle represents a 2-item set consisting of A1 and A2. If the 2-item set meets the minimum support count value for sample particle swarm, the certain candidate particle would be reserved or removed conversely.

(iii) Rule particle swarm: in the APPSO algorithm, a particle swarm is randomly created (corresponding to the second link in the Apriori algorithm) such that each particle of the particle swarm represents a potential association rule. The length of each particle is equal to the length of each particle in the candidate particle swarm. Each alarm is expressed with 0 or 1. 1 indicates the corresponding alarm is the antecedent of the association rule while 0 indicates that the corresponding alarm is the consequent of the association rule. Such a particle swarm is referred to as rule particle swarm C (RPS-C).

Assume the value of a certain particle  $b$  in particle swarm C is 111000 and then the rule represented is (A1, A2, A3)  $\Rightarrow$  (A4, A5, A6).

After creating of candidate particle swarm B and rule particle swarm C, the operational method for the two particle

swarms is as follows (particle  $a$  belongs to candidate particle swarm B and particle  $b$  belongs to rule particle swarm C).

The logic operation of “and” is performed for each particle of candidate particle swarm B and each particle of rule particle swarm C and the operational result is used to estimate the relation between the antecedent and consequent of the rule. For example,  $a = 110011$ ,  $b = 111000$ , and  $a \cap b = 111000$  indicate that Alarm A3 and Alarm A4 are not in the association rules. The field value of A2 and A2 is 1 and the field value of A4 and A6 is 0. We can obtain that the association rule represented by  $a$  and  $b$  is (A1, A2)  $\Rightarrow$  (A5, A6).

**5.3.2. APPSO Algorithm Optimization Link.** During mining of association rules based on swarm intelligence, the particle ergodic method is usually used to obtain the support of the item set represented by the particle. The particle support obtained by scanning the whole database is accurate in result. However, some shortcomings exist; that is, the actual analysis efficiency is low and no data source characteristics and basic algorithm characteristics are combined. Therefore, data source sequencing coding and sliding window value assignment are used based on the data characteristics of the network alarms; the sparse linked list algorithm is deployed to calculate the support of the item set.

(1) *Sequencing Code.* As alarm names are usually described with English character string or digit combined number, such an identification method would bring about a large amount of combined data (e.g. MPLS\_TUNNEL\_MISMERGE and 007-061-00-800446) resolution consumption to data processing and analysing. Therefore, we employ the method by sequencing codes to reduce resolution consumption, in which all alarm names or network management alarm IDs are sequenced on the basis of the sequence of letters and figures. It targets on avoiding two or more integral values being assigned to the same alarm subsequently (Figure 5); differentiated values are assigned on the basis of data sequence.

(2) *Sliding Window.* Due to the combination of time-type data and relationship type in alarms, the time-type alarm data is sequenced on the basis of time length, the size of sliding time window, and sliding step length, and the relationship type alarm data is converted and combined into different transactional data item sets.

(3) *Sparse Linked List.* Compared with the overall alarm database after division, each of the alarm data item sets only contains partial alarm data types. The efficiency of database scanning by the APPSO algorithm is further improved using the thought of sparse linked list based on the data characteristics. The algorithm process is as follows.

A linked list header is created for each item of the whole database. For example, if there are 200 alarm code integer data types, consisting in 10,000 item sets, 200 linked list headers will be created and the integral value of each item is the number of its corresponding linked list.

The item sets are scanned in sequence and the items of each item set are added to the end of the corresponding linked list. For example, If the  $n$ th item set in the database is (50, 108,

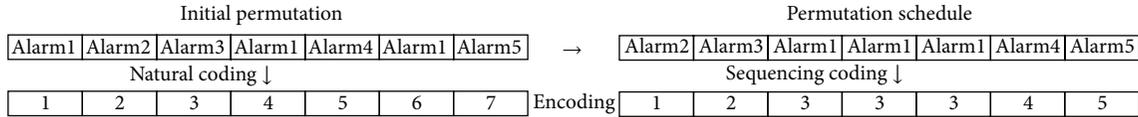


FIGURE 5: Natural coding and sequencing coding.

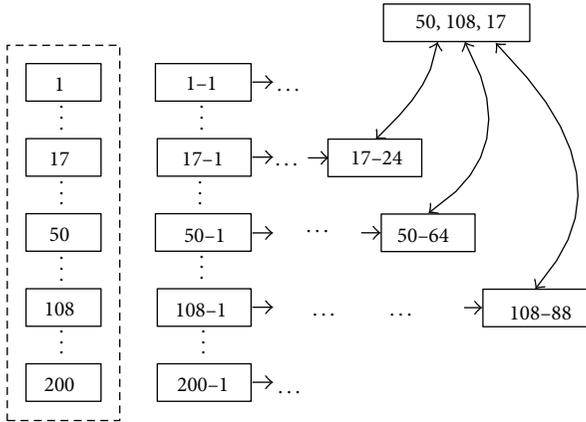


FIGURE 6: Examples of sparse linked list.

17), then the  $n$ th item set is added to the end of the linked list 50 and the end of the linked list 108, and so forth. 200 linked lists are created finally, that is, sparse linked list. The number of the alarm code integers saved in each linked list is much less than the 10000 item sets of the whole database (Figure 6).

(4) *Calculation of the Particle Support Based on the Sparse Linked List.* Take the  $n$ th item set in the database (50, 108, 17) and 200 linked list headers as examples (Figure 7).

Starting with the linked list 50, it is assumed to be containing the item “50” through searching the 64th item set. Similarly, the linked lists 108 and 17 correspond to 88 and 24, respectively; that is, all item sets before the 88th item set do not contain the corresponding item of the particle. After searching in the 88th item set, 1 will be added to the particle support if it contains (50, 108, 17) (Step 1), otherwise, continually searching in the linked list (50, 108, 17) in order to find the next data, respectively. Assume that they correspond to 121, 90, and 65, respectively, and directly search in the 121st item set. 1 will be added to the particle support if it contains (50, 108, 17) (Step 2); otherwise, continue to search in the linked list (50, 108, 17) and find the next data. Suppose that they correspond to 121, 184, and 121, respectively, and directly search in the 184th item set. 1 will be added to the particle support if it contains (50, 108, 17) (Step 3); otherwise, keep on searching. The overall linked list would finish searching when 50 has been sorted out in (50, 108, 17) (Step 4).

(5) *Nature of the Apriori Algorithm.* Based on the nature of the Apriori algorithm: “the subset of the known frequent item set  $k$  is also frequent;” the nature is used to optimize the

search rule for the particle swarm; that is, all subsets of the particle are also frequent if the corresponding candidate item set of a certain particle is a frequent item set. For example, if the particle  $a$  (110011) belongs to a frequent item set, then any subset of the value of  $a$ , such as 110000, 000011, 100001, 010010, 100010, and 010001, are frequent and these subsets are directly incorporated into candidate particle swarm A as new particles.

In conclusion, the main principle of the APPSO algorithm is to estimate whether each particle in candidate particle swarm A (CPS-A) is frequent or not. The subset of the particle will be added to A if the particle is frequent. Then the logical operation of “and” is performed for the particle and each particle of rule particle swarm B (RPS-B) to judge whether the corresponding rule of the result obtained is an association rule meeting the conditions or not. In accordance with a certain sequence, A and B are constantly updated until all iterative processes terminate.

5.3.3. *APPSO Algorithm Test.* A comparison test is conducted on the test platform with the APPSO algorithm and Apriori algorithm (hardware: CPU Intel, Core i5 3.3 GHz, 8 G RAM, 1 T hard disk, software: operating system window7, development platform Qt4.7.0, single-thread development). The alarm data (21084 pieces) of the network management system PTN device is extracted at random as the data. The data is generated into item sets with 5-seconds (5 s) time window and the data set containing only a single item (1-item sets) is rejected. Finally, 4753 item sets in total are obtained. The scales of candidate particle swarm and the rule particle swarm are identical.

(i) Test 1: relation between the support and number of association rules: the scale of the particle swarm is 40, number of iterations is 100, and confidence coefficient is 30%.

Analysis on Test 1: Apriori algorithm is a global search algorithm. Therefore, the number of the association rules mined by the APPSO algorithm is less than the number of the association rules mined by the Apriori algorithm. More than 60% of the main association rules is obtained with the APPSO algorithm as shown in Figure 8.

(ii) Test 2: relation between the confidence coefficient and number of association rules: the scale of the particle swarm is 40, number of iterations is 100, and confidence coefficient is 5%.

Analysis on Test 2: under the condition of a constant number of iterations and minimum support, the number of alarms obtained by the two algorithms will necessarily decrease with increasing of confidence coefficient index; compared with the Apriori algorithm, when the confidence

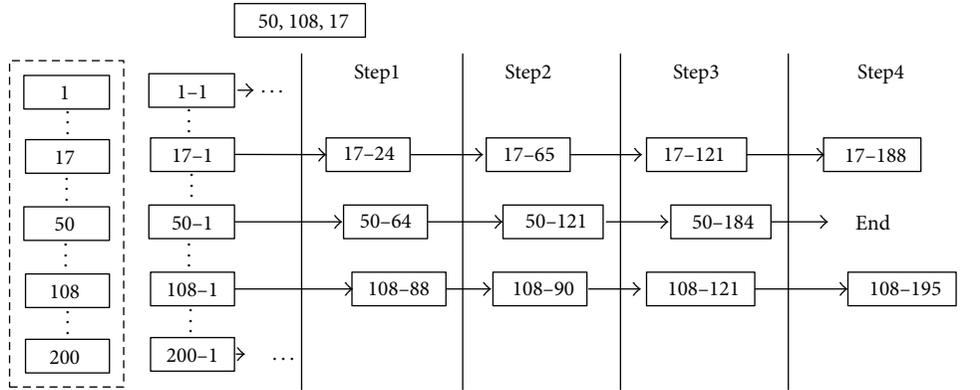


FIGURE 7: Examples of calculation of the particle support based on the sparse linked list.

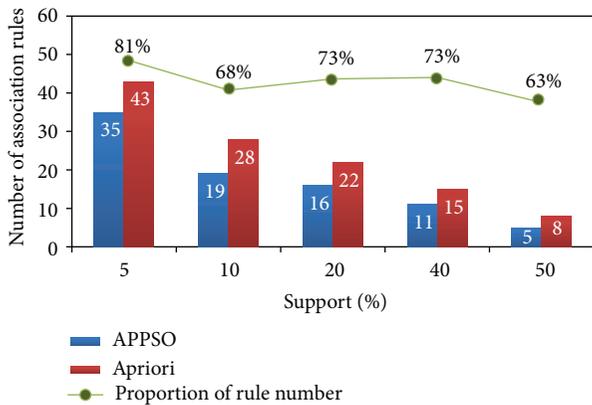


FIGURE 8: Relation between the support and number of association rules.

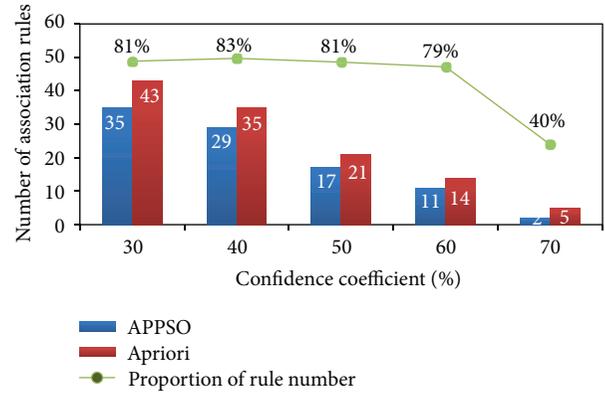


FIGURE 9: Relation between the confidence coefficient and number of association rules.

coefficient value is within the discrete interval [30%, 60%], the number of the association rules obtained with the APPSO algorithm accounts for approximately 80% as shown in Figure 9.

(iii) Test 3: relation between the scale of the particle swarm and the number of association rules: the number of iterations is 100, the minimum support is 5%, and confidence coefficient is 30%.

Analysis on Test 3: under the condition of a constant number of iterations, minimum support, and confidence coefficient, the larger the particle swarm is, the more the number of the association rules will be. The number of the association rules will approach the number of the rules obtained by the global search of the Apriori algorithm as shown in Figure 10.

(iv) Test 4: relation between the number of iterations and operation time: the scale of the particle swarm is 40, minimum support is 5%, and the confidence coefficient is 30%.

Analysis on Test 4: under the condition of a constant particle swarm scale, minimum support, and confidence coefficient, the time for the APPSO algorithm is prolonged

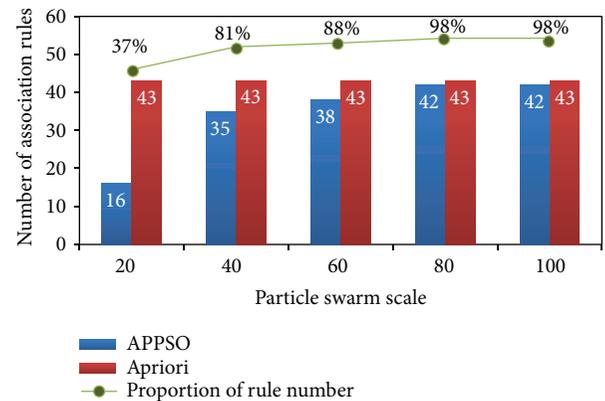


FIGURE 10: Relation between the scale of the particle swarm and the number of association rules.

with increase of the number of iterations but the number of association rules obtained significantly increases; compared with the Apriori algorithm, the efficiency of the APPSO algorithm significantly increases; for example, the number of iterations is 120, the time for the APPSO algorithm only

TABLE 9: Distribution of training data sets.

Item sets	1-item sets	2-item sets	3-item sets	4-item sets	5-item sets	6-item sets	7-item sets	8-item sets	9-item sets	10-item sets
Number of item sets	136342	91787	36780	10141	2578	626	206	109	188	204
Item sets	11-item sets	12-item sets	13-item sets	14-item sets	15-item sets	16-item sets	17-item sets	18-item sets	20-item sets	100-item sets
Number of item sets	194	704	343	239	186	54	30	6	0	1

TABLE 10: Distribution of test data sets.

Item sets	1-item sets	2-item sets	3-item sets	4-item sets	5-item sets	6-item sets	7-item sets	8-item sets	9-item sets
Number of item sets	15455	8649	1386	232	33	5	5	4	8
Item sets	10-item sets	11-item sets	12-item sets	13-item sets	14-item sets	15-item sets	16-item sets	17-item sets	18-item sets
Number of item sets	1	7	55	10	10	9	4	1	1

TABLE 11: Statistics on association rate of test data.

Minimum support count	0.01	0.01	0.01
Minimum confidence	0.01	0.05	0.09
Number of rules from training data sets	185	149	154
Alarm association rate from test data sets	81.23%	81.21%	81.22%

accounts for 17% of the time for the Apriori algorithm yet the number of the rules obtained accounts for 88% of the total number of rules as shown in Figure 11.

On the premise of desired demand for the number of rules, the APPSO algorithm is able to control the operational

precision and decrease the computation time and memory consumption by reasonably setting the particle swarm parameters.

(v) Engineering test: the network alarm data over the four of the 8 consecutive weeks is used as "training data." The alarm association rules are mined by the APPSO algorithm and the data over the other 4 weeks is used as "test data" to calculate the alarm correlation rate. Specific method: all alarms are intercepted as per the fixed flow time window and all of the non-1-item sets are included in the calculation of the alarm correlation rate (the 1-item sets themselves do not have a correlation relationship). The algorithm is as follows:

$$\text{Alarm association rate} = \left( \frac{\text{number of non-1-item sets meeting the association rules}}{\text{number of all non-1-item sets}} \right) \times 100\%. \quad (1c)$$

For example, the alarm sequence is (A1, A2, A3, A1, A4, A2, A3, A4, A4, A2), and becomes {A1, A2}, {A3}, {A1, A4, A2}, {A3, A4}, {A4, A2} after being intercepted in accordance with the fixed flow time window, among which the non-1-item sets involving in the calculation of the alarm correlation rate are {A1, A2}, {A1, A4, A2}, {A3, A4}, and {A4, A2}. The association rate of the alarm data is 50% if the association rule is A1 → A2.

Analysis on engineering test: The alarm association rules obtained through the training data over the first 4 weeks is applied in the test data over the last 4 weeks. The training data over the first 4 weeks contains the equipment types, BSC, BTS, CELL, and 516271 alarms, of which the alarm types are 131. The time window is set to 2 s and the sliding step length to 1 s; the test data over the last 4 weeks contains the equipment types, BSC, BTS, CELL, and 39470 alarms, of which the alarm types are 89. In combination with the requirements for actual

conditions of the engineering operating environment, the time window is set to 3 s. 10420 non-1-item sets are obtained after interception of data.

From Tables 9, 10, and 11 it is obtained that all of the alarm association rates are higher than 80%. The APPSO association mining algorithm provides an effective analytic method for the alarm association analysis.

## 6. Conclusion

The association rules for the alarm data in the information communication network should be analysed in conjunction with the data characteristics to perform a design specifically to achieve a corresponding algorithm flow. Compared with the Apriori algorithm, the mining efficiency of the APPSO algorithm is significantly enhanced but a small number of

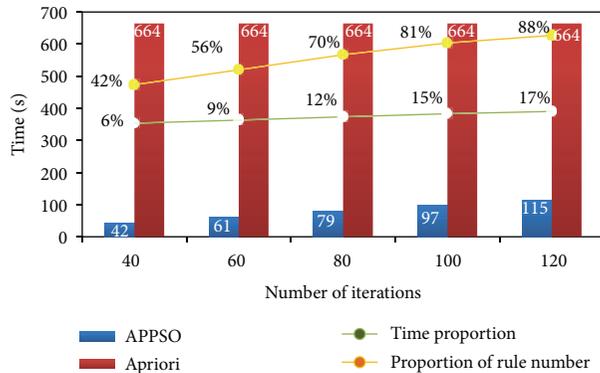


FIGURE 11: Relation between the number of iterations and operation time.

association rules are lost to some extent due to the characteristics of the PSO algorithm. The value of the association rules lies in quick acquisition and subsequent high-value evaluation of association logic instead of sole acquisition of all association rules. From this perspective, the APPSO algorithm improves in both mining efficiency and algorithm concept.

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## Acknowledgments

This research was supported by a grant from the National Natural Science Foundation of China (no. 51205274), the Science and Technology Major Project of the Shanxi Science and Technology Department (20121101004), the Key Disciplines Construction in Colleges and Universities of ShanXi ([2012]45), the Shanxi Scholarship Council of China (no. 2013-035), the China Postdoctoral Science Foundation (no. 2013M530894), and the Innovation Project of the Postgraduate Education in Shanxi Province (no. 20123027).

## References

- [1] E. Kiciman and A. Fox, "Detecting and localizing anomalous behavior to discover failures in component-based internet services," Tech. Rep., The Stanford Computer Science (CS) Department, Stanford, Calif, USA, 2004.
- [2] L. L. Huang, G. G. Su, and Y. J. Jiang, *Operation Support System Technology and Practice*, Posts & Telecom Press, Beijing, China, 2012 (Chinese).
- [3] D. T. Li, *Researches on data mining based alarm correlation analysis in communication networks [Ph.D. thesis]*, University of Electronic Science and Technology of China, Chengdu, China, 2010 (Chinese).
- [4] B. Z. Yao, R. Mu, and B. Yu, "Swarm intelligence in engineering," *Mathematical Problems in Engineering*, vol. 2013, Article ID 835251, 3 pages, 2013.
- [5] Y. Wang, G. C. Li, and Y. K. Xu, "Research on management method, classification and correlation of alarm in information communication network," *Telecommunications Science*, vol. 29, no. 8, pp. 132–135, 2013 (Chinese).
- [6] X. B. Wang, W. Li, and H. W. Xu, "Management analysis of alarm standardization in centralized operational mode," *Telecommunications Technology*, no. 4, pp. 39–42, 2009 (Chinese).
- [7] T.-Y. Li and X.-M. Li, "Preprocessing expert system for mining association rules in telecommunication networks," *Expert Systems with Applications*, vol. 38, no. 3, pp. 1709–1715, 2011.
- [8] H. Mannila, H. Toivonen, and I. Verkamo, "Discovery of frequent episodes in event sequences," *Data Mining and Knowledge Discovery*, vol. 1, no. 3, pp. 259–289, 1997.
- [9] R. Sterritt, D. Bustard, and A. McCrea, "Autonomic computing correlation for fault management system evolution," in *Proceedings of the IEEE International Conference Industrial Informatics (INDIN '03)*, pp. 240–247, Alberta, Canada.
- [10] A. A. Amaral, B. Z. Zarpelão, L. M. Mendes et al., "Inference of network anomaly propagation using spatio-temporal correlation," *Journal of Network and Computer Applications*, vol. 35, no. 6, pp. 1781–1792, 2012.
- [11] X. D. Wu and K. Vipin, *The Top Ten Algorithms in Data Mining*, Chapman and Hall/CRC, Boca Raton, Fla, USA, 2009.
- [12] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules in large databases," in *Proceedings of the 20th International Conference on Very Large Data Bases*, pp. 487–499, Santiago de Chile, Chile, 1994.
- [13] S. Y. Jiang, X. Li, and Q. Zheng, *Principles and Practice of Data Mining*, Publishing House of Electronics Industry, Beijing, China, 2011 (Chinese).
- [14] T. Calders, N. Dexters, J. J. M. Gillis, and B. Goethals, "Mining frequent itemsets in a stream," *Information Systems*, vol. 39, pp. 233–255, 2012.
- [15] V. D. Mabonzo, *Study on new approach for effective mining association rules from huge databases [Ph.D. thesis]*, Dalian Maritime University, Dalian, China, 2012.
- [16] K. Z. Ziauddin, K. T. Shahid, and Z. K. Khaiuz, "Research on association rule mining," *Advances in Computational Mathematics and Its Applications*, vol. 2, no. 1, pp. 226–236, 2012.
- [17] J. W. Han, J. Pei, and Y. W. Yin, "Mining frequent patterns without candidate generation," in *Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD '00)*, pp. 1–12, Dallas, Tex, USA, 2000.
- [18] X. S. Yang, Z. H. Cui, R. B. Xiao et al., *Swarm Intelligence and Bio-Inspired Computation: Theory and Applications*, Elsevier, Amsterdam, The Netherlands, 2013.
- [19] C. W. Reynolds, "Flocks, herds, and schools: a distributed behavioral model," *Computer Graphics*, vol. 21, no. 4, pp. 25–34, 1987.
- [20] J. Kennedy and R. C. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, pp. 1942–1948, December 1995.
- [21] G. Veysel and M. P. Kevin, *Swarm Stability and Optimization*, Springer, Berlin, Germany, 2011.

## Research Article

# Hybrid Functional-Neural Approach for Surface Reconstruction

Andrés Iglesias<sup>1,2</sup> and Akemi Gálvez<sup>1</sup>

<sup>1</sup> Department of Applied Mathematics and Computational Sciences, E.T.S.I. Caminos, Canales y Puertos, University of Cantabria, Avenida de los Castros, s/n, 39005 Santander, Spain

<sup>2</sup> Department of Information Science, Faculty of Sciences, Toho University, 2-2-1 Miyama, Funabashi 274-8510, Japan

Correspondence should be addressed to Andrés Iglesias; iglesias@unican.es

Received 28 July 2013; Accepted 8 December 2013; Published 16 January 2014

Academic Editor: Yudong Zhang

Copyright © 2014 A. Iglesias and A. Gálvez. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper introduces a new hybrid functional-neural approach for surface reconstruction. Our approach is based on the combination of two powerful artificial intelligence paradigms: on one hand, we apply the popular Kohonen neural network to address the data parameterization problem. On the other hand, we introduce a new functional network, called NURBS functional network, whose topology is aimed at reproducing faithfully the functional structure of the NURBS surfaces. These neural and functional networks are applied in an iterative fashion for further surface refinement. The hybridization of these two networks provides us with a powerful computational approach to obtain a NURBS fitting surface to a set of irregularly sampled noisy data points within a prescribed error threshold. The method has been applied to two illustrative examples. The experimental results confirm the good performance of our approach.

## 1. Introduction

Manufacturing industries are constantly evolving in response to the new challenges of the globalization and the growing competition in this global market. Product design is playing a central role in this process, as current customers are increasingly demanding a mass customization of the products. As a result, the geometric and aesthetic properties of the manufactured goods (shape, color, and dimensions) have to be modified frequently in order to meet the new market demands.

A major step in this process is the generation of real prototypes with different materials to explore and analyze their geometric properties and the feedback of potential customers when exposed to different variations of the final product. Prototype generation and customization can be dramatically improved by using digital technologies, in which the physical model is digitized, stored, and manipulated by computer, a process called reverse engineering [1, 2]. Typically, this process begins with data sampling by using 3D laser scanning and other digitizing devices. This technology is intensively used for the construction of car bodies, ship hulls, airplane fuselage, and other free-form objects [2–7]. The resulting data

points are then fitted to mathematical entities such as curves and surfaces, usually in parametric form. The output is a very accurate digital version of the real product, which is also simpler and easier to store, analyze, and manipulate. It also simplifies the transfer and communication processes among designers, manufacturers, and providers, making the model available in just a few seconds all over the world, a key aspect in our ubiquitously connected information society era.

In this paper, we are interested in one of the most critical steps of this process, namely, the construction of surfaces of the real objects from sets of digitized data points, a field usually called *surface reconstruction*. The preferred mathematical models for design and manufacturing are the free-form parametric surfaces [3, 8–16], because they are very flexible and can be readily modified by changing a small set of parameters. In this paper, we use NURBS surfaces, the most powerful (and most difficult to deal with) free-form parametric surfaces, which have become the standard for CAD/CAM data representation in industrial settings and in many other fields, from digital effects for movies in computer animation and advertisements to the design of characters in computer graphics and video games. In reverse engineering applications, data points are usually acquired through laser

scanning and other digitizing devices and are, therefore, subjected to measurement noise, irregular sampling, and other artifacts [6, 7]. Consequently, a good fitting of data should be generally based on approximation schemes rather than interpolation [1, 13, 14, 17–19]. Because this is the typical case in many real-world industrial problems, in this paper we focus on the approximation scheme to a given set of noisy, irregularly sampled data points.

Obtaining the best approximating surface in such cases is much more troublesome than it may seem at first sight. The main reasons are as follows.

- (i) The NURBS surfaces depend on many different parameters (data parameters, knots, control points, and weights) that are strongly interconnected with each other, leading to a strongly nonlinear continuous optimization problem.
- (ii) It is also multivariate, as it typically involves a large number of unknown variables for a large number of data points, a case that happens very often in real-world examples.
- (iii) In addition, it is also overdetermined, because we expect to obtain the approximating surface with many fewer parameters than the number of data points.
- (iv) Finally, the problem is known to be multimodal; that is, the least-squares objective function can exhibit many local optima [20, 21], meaning that the problem might have several (global and/or local) good solutions.

In conclusion, we have to solve a very difficult multimodal, multivariate, high-dimensional continuous nonlinear optimization problem. A number of methods have been proposed to solve this problem (see Section 2 for details). Among them, those based on artificial intelligence techniques have received increasing attention during the last few years. Most of such methods rely on the *artificial neural networks* (ANN) formalism [22]. Since ANN methodology is actually inspired by the behavior of the human brain, it is able to reproduce some of its most typical features, such as the ability to learn from data. This explains why they have been so widely applied to data fitting problems.

Although they are very popular, the ANN are however limited in many aspects. A major drawback is their inability to reproduce mathematically the functional structure of a given problem. This limitation can be overcome with the use of a new paradigm in artificial intelligence, the so-called *functional networks* (FN) (see Section 4 for details). In short, functional networks are a generalization of the standard neural networks in which the scalar weights are replaced by neural functions. These neural functions can exhibit, in general, a multivariate character. Furthermore, different neurons can be associated with neural functions from different families of functions. These FN features allow us to reproduce exactly the functional structure of the problem by a careful choice of the functions involved, which can hereby be associated with one or several neurons of the network. This procedure yields a functional structure that is

typically a replica of the underlying structure of the given problem.

*1.1. Aims and Structure of the Paper.* In this paper, we propose a hybrid artificial intelligence approach to solve the surface reconstruction problem. Our approach is based on the combination of two powerful artificial intelligence paradigms: on one hand, we apply the popular Kohonen neural network to address the data parameterization problem. On the other hand, we take advantage of the remarkable properties mentioned in previous paragraph by introducing a new functional network, called NURBS functional network, whose topology is specially targeted to reproduce the functional structure of the NURBS surfaces. These neural and functional networks are then applied iteratively for further surface refinement. As it will be shown later on, the hybridization of these two networks provides us with a powerful computational approach to solve the surface reconstruction problem. To check the performance of our approach, it has been applied to two illustrative examples. Our experimental results show that the method performs very well, being able to reconstruct the approximating surface of the given set of data points with a high degree of accuracy.

The structure of this paper is as follows: in Section 2, previous work regarding the surface reconstruction problem is reported. Then, some basic concepts about NURBS surfaces and the optimization problem to be solved are given in Section 3. Section 4 describes the fundamentals of the functional networks along with their main components and the differences between neural and functional networks. The proposed hybrid functional-neural method for surface reconstruction with NURBS surfaces is described in detail in Section 5. Then, some illustrative examples of its application are reported in Section 6. A comparison of our approach with other ANN alternative methods is analyzed in detail in Section 7. The paper closes with the main conclusions of this contribution and our plans for future work in the field.

## 2. Previous Work

Surface reconstruction has been a topic of increasing attention from the scientific community during the last 20 years, with outstanding applications in both theoretical and applied domains. Regarding the theoretical side, it is a remarkable subject in approximation theory [23, 24], statistics [25], numerical analysis [26, 27], geometric modeling [2, 8, 28], and computer-aided geometric design (CAGD) [5, 29]. In addition, there is a bulk of applications in several fields, such as computer-aided manufacturing (CAM) [6, 7], data visualization [30], cultural heritage preservation [31], virtual reality [32], medical imaging [33], and computer animation [34], to mention just a few.

In general, surface reconstruction methods are classified in terms of the available input (2D slices, isoparametric curves, clouds of points, mixed information, etc.). For instance, authors in [35–39] address the problem of obtaining a surface model from a set of given cross-sections, a classical problem in medical science, biomedical engineering, and

CAD/CAM. Other classical input data include isoparametric curves on the surface [40] and even mixed information, such as scattered points and contours [41–43] or isoparametric curves and data points [4, 5, 44].

In most cases, however, the available information about the surface is typically a dense set of (usually unorganized) 3D data points obtained by using some sort of digitizing devices (see, e.g., [14, 45–48]). In that case, the reconstructed surface can be described using three different representations providing different levels of accuracy. The simplest one is given by the polygonal meshes, where the data points are used as vertices connected by lines (edges) that work together to create a 3D model, comprised of vertices, edges, and faces. Although it is the coarsest representation, it is also the most popular because of its simplicity, flexibility, and excellent performance with current graphical cards. Surface reconstruction methods with polygonal meshes can be found, for instance, in [30, 31, 47, 49–51] and references therein. The next level is given by the constructive solid geometry (CSG) models, where elementary geometries (such as spheres, boxes, cylinders, or cones) are combined in order to produce more elaborated shapes by applying some simple (Boolean) operators: union, intersection, and difference. This methodology works well but presents a low level of flexibility, being severely limited to very simple shapes. The most sophisticated and most accurate level consists of obtaining the real mathematical surface fitting the data points. This issue has been analyzed from several points of view, such as parametric methods [52], subdivision surfaces [53], function reconstruction [54, 55], implicit surfaces [48], and algebraic surfaces [56]. Other approaches are based on the application of metaheuristic techniques, which have been intensively applied to solve difficult optimization problems that cannot be tackled through traditional optimization algorithms. Recent schemes in this area involve particle swarm optimization [10, 57, 58], genetic algorithms [59–62], artificial immune systems [63, 64], estimation of distribution algorithms [65], firefly algorithm [66, 67], and hybrid techniques [68, 69].

Artificial neural networks have also been applied to this problem [45, 70], mostly for arranging the input data in case of unorganized points. After this preprocessing step, any other classical surface reconstruction method operating on organized points is subsequently applied. A work using a combination of neural networks and partial differential equation (PDE) techniques for the parameterization and reconstruction of surfaces from 3D scattered points can be found in [71]. Two previous papers by the authors have also addressed this problem by using functional networks [4, 8], a powerful generalization of neural networks based on functional equations [72, 73]. Both works show, however, that the single application of functional networks is still unable to solve the general case. The work in [4] addresses the particular case of B-spline surface reconstruction when some additional information (isoparametric curves) is available in addition to the data points. The fitting surfaces are a generalization of the Gordon surfaces [40]. The work in [8] combines functional networks with genetic algorithms in order to solve the (much simpler) polynomial Bézier case. The approach presented here solves the general NURBS

surface reconstruction problem based exclusively on neural and functional networks. To the best of our knowledge, no previous method is reported in the literature providing all these features.

### 3. Basic Concepts and Definitions

**3.1. NURBS Surfaces.** Let  $\mathcal{S} = \{s_0, s_1, s_2, \dots, s_{r-1}, s_r\} \subset [a, b]$  be a nondecreasing sequence of real numbers called *knots*.  $\mathcal{S}$  is called the *knot vector*. Without loss of generality, we can assume that  $[a, b] = [0, 1]$ . The *i*th *B-spline basis function*  $N_{i,k}(u)$  of order *k* (or equivalently, degree  $k - 1$ ) is defined by the recurrence relations:

$$N_{i,1}(u) = \begin{cases} 1 & \text{if } s_i \leq u < s_{i+1} \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

with  $i = 0, \dots, r - 1$  and

$$N_{i,k}(u) = \frac{u - s_i}{s_{i+k-1} - s_i} N_{i,k-1}(u) + \frac{s_{i+k} - u}{s_{i+k} - s_{i+1}} N_{i+1,k-1}(u) \quad (2)$$

for  $k > 1$ . Note that *i*th B-spline basis function of order 1,  $N_{i,1}(u)$ , is a piecewise constant function with value 1 on the interval  $[s_i, s_{i+1})$ , called the *support* of  $N_{i,1}(u)$ , and zero elsewhere. This support either can be an interval or reduce to a point, as knots  $s_i$  and  $s_{i+1}$  must not necessarily be different. If necessary, the convention  $0/0 = 0$  in (2) is applied. Any basis function of order  $k > 1$ ,  $N_{i,k}(u)$ , is a linear combination of two consecutive functions of order  $k - 1$ , where the coefficients are linear polynomials in  $u$ , such that its order (and hence its degree) increases by 1. Simultaneously, its support is the union of the (partially overlapping) supports of the former basis functions of order  $k - 1$  and, consequently, it usually enlarges.

With the same notation, given a set of three-dimensional points (called *control points* as they roughly determine the shape of the curve)  $\{\mathbf{P}_{ij}\}_{i=0,\dots,m;j=0,\dots,n}$  in a bidirectional net and two knot vectors  $\mathcal{S} = \{s_0, s_1, s_2, \dots, s_{r-1}, s_r\}$  and  $\mathcal{T} = \{t_0, t_1, \dots, t_{h-1}, t_h\}$ , a *NURBS surface*  $\mathbf{S}(u, v)$  of order  $(k, l)$  (where *NURBS* stands for Non-Uniform Rational B-Spline) is a rational B-spline parametric surface given by

$$\mathbf{S}(u, v) = \frac{\sum_{i=0}^m \sum_{j=0}^n w_{i,j} \mathbf{P}_{i,j} N_{i,k}(u) N_{j,l}(v)}{\sum_{i=0}^m \sum_{j=0}^n w_{i,j} N_{i,k}(u) N_{j,l}(v)}, \quad (3)$$

where the  $\{N_{i,k}(u)\}_i$  and  $\{N_{j,l}(v)\}_j$  are the B-spline basis functions of orders  $k$  and  $l$ , respectively, defined following (1) and (2), and  $\{w_{i,j}\}_{i,j}$  are nonnegative scalar values called weights associated with the control points  $\{\mathbf{P}_{i,j}\}_{i,j}$ . Without loss of generality, parameters  $u, v$  can be assumed to take values on the interval  $[0, 1]$ . For a proper definition of a NURBS surface in (3), the following relationships must hold (see [29]):  $r = m + k, h = n + l$ .

In general, a NURBS surface does not interpolate any of its control points; the interpolation only occurs for nonperiodic knot vectors (in that case, the NURBS surface does interpolate the corner control points) [11, 29]. Since they are the

most common in computer graphics and industrial domains, in this work we will consider the case of nonperiodic knot vectors. Note, however, that our method does not depend on the kind of knot vectors used for the approximating surfaces.

**3.2. Surface Reconstruction Problem.** In clear contrast with many previous methods, in this paper we focus on the *general* surface reconstruction problem, which assumes that no other information about the problem is available beyond the data points. In particular, our problem can be stated as follows. Given a set of (usually irregularly sampled) noisy data points  $\mathbf{Q}$  assumed to lie on an unknown surface  $\mathbf{U}$ , construct, to the extent possible, a full mathematical representation of a surface model  $\mathbf{S}$  that approximates  $\mathbf{U}$ . Because of its remarkable applications in real-world engineering problems, we also demand such a mathematical representation to be a NURBS surface.

Mathematically speaking, we assume that we are provided with a set of data points  $\{\mathbf{Q}_{\alpha,\beta}\}_{\alpha=1,\dots,M;\beta=1,\dots,N}$  in  $\mathbb{R}^3$ , with  $M, N \gg r, h$ . Our goal is to obtain the NURBS surface  $\mathbf{S}(u, v)$  that fits the data points better in the discrete least-squares sense. To do so, we have to compute all the parameters of the approximating surface by minimizing the least-squares error,  $E$ , defined as the sum of squares of the residuals:

$$E = \sum_{\alpha=1}^M \sum_{\beta=1}^N \left( \mathbf{Q}_{\alpha,\beta} - \frac{\sum_{i=0}^m \sum_{j=0}^n w_{i,j} \mathbf{P}_{i,j} N_{i,k}(u_\alpha) N_{j,l}(v_\beta)}{\sum_{i=0}^m \sum_{j=0}^n w_{i,j} N_{i,k}(u_\alpha) N_{j,l}(v_\beta)} \right)^2. \quad (4)$$

In the case of scattered data points  $\{\mathbf{Q}_\mu\}_{\mu=1,\dots,R}$ , our method will work in a similar way by simply replacing the previous expression (4) by

$$E = \sum_{\mu=1}^R \left( \mathbf{Q}_\mu - \frac{\sum_{i=0}^m \sum_{j=0}^n w_{i,j} \mathbf{P}_{i,j} N_{i,k}(u_\mu) N_{j,l}(v_\mu)}{\sum_{i=0}^m \sum_{j=0}^n w_{i,j} N_{i,k}(u_\mu) N_{j,l}(v_\mu)} \right)^2. \quad (5)$$

The minimization of either (4) or (5) leads to the system of equations:

$$\mathbf{Q}^v = \mathbf{M} \cdot \mathbf{P}^v, \quad (6)$$

where the symbol  $(\cdot)^v$  denotes the vectorization operator of the given matrix and  $\mathbf{M}$  in (6) is a matrix given by  $\mathbf{M}_{i,j}(u_\alpha, v_\beta) = [((\mathbf{R}_{j,l}^v(u_\alpha, v_\beta))^T \otimes \mathbf{R}_{i,k}(\mathbf{u}^T, v_\beta))^v]^T$ , with  $\mathbf{u} = (u_1, \dots, u_M)$ .  $R_{i,j}(u, v)$  is given as

$$R_{i,j}(u, v) = \frac{w_{i,j} N_{i,k}(u) N_{j,l}(v)}{\sum_{i=0}^m \sum_{j=0}^n w_{i,j} N_{i,k}(u) N_{j,l}(v)} \quad (7)$$

$$i = 0, \dots, m; \quad j = 0, \dots, n$$

while  $\otimes$  and  $(\cdot)^T$  represent the outer product operator and the transpose of a vector or matrix, respectively. The indices in (4)–(7) vary in the ranges of values indicated throughout the section.

It is worthwhile to mention that since the lengths of  $\mathbf{Q}^v$  and  $\mathbf{P}^v$  are, respectively,  $M \times N$  and  $(m+1) \times (n+1)$ ,

the system (6) is overdetermined. Premultiplication of both sides by  $\mathbf{M}^T$  yields

$$\mathbf{M}^T \cdot \mathbf{Q}^v = \mathbf{M}^T \cdot \mathbf{M} \cdot \mathbf{P}^v = \mathbf{\Delta} \cdot \mathbf{P}^v, \quad (8)$$

where  $\mathbf{\Delta} = \mathbf{M}^T \cdot \mathbf{M}$ . Note also that the tensor-product basis functions  $R_{i,j}(u, v)$  are generally continuous and nonlinear, so the minimization of (8) leads to the continuous nonlinear optimization problem:

$$\min_{\{\mathbf{P}_{i,j}\} \subset \mathbb{R}^3; \{w_{i,j}\} \geq 0; \{(u_\alpha, v_\beta)\} \subset \text{Dom}(\mathbf{S}); \{s_v\}, \{t_v\} \subset [0,1]} \|\mathbf{M}^T \cdot \mathbf{Q}^v - \mathbf{\Delta} \cdot \mathbf{P}^v\|^2, \quad (9)$$

where  $\|\cdot\|$  represents the Euclidean norm.

## 4. Functional Networks

Roughly speaking, a *functional network* is a generalization of the standard neural network in which the scalar weights are replaced by neural functions. Functional networks were firstly introduced in 1998 by Castillo in [72] as a way to enhance the neural networks with new capabilities. Since then, they have been successfully applied to several problems in science and engineering. The interested reader is referred to [73, Chapter 9] (Chapter 9) for in-depth explanation about functional networks along with several illustrative examples and applications. In this section we describe the main components of a functional network. Differences between neural and functional networks are also discussed in this section.

**4.1. Components of a Functional Network.** As an explanatory example, Figure 1(a) shows the functional network of the associative operation  $F$  between two real numbers; that is, function  $F$  satisfies

$$F(F(x, y), z) = F(x, F(y, z)). \quad (10)$$

It can be proved that the general solution of this equation is given by [73]

$$F(x, y) = f^{-1}[f(x) + f(y)], \quad (11)$$

where  $f(x)$  is an arbitrary continuous and strictly monotonic function, which can be replaced only by  $c f(x)$ , where  $c$  is an arbitrary constant. Such a solution can be represented by the functional network in Figure 1(b). Note that, because of the uniqueness (except arbitrary constants) of the solution, both networks do actually represent the same problem. In other words, the functional network in Figure 1(b), is equivalent to (but arguably simpler than) that in Figure 1(a). From Figure 1(b) the main components of a functional network become clear.

(i) *Several Layers of Storing Units*

(a) *A Layer of Input Units.* This first layer contains the input information. In this figure, this input layer consists of the units  $x$  and  $y$ .

(b) *A Set of Intermediate Layers of Storing Units.* They are not neurons but units storing intermediate information. This

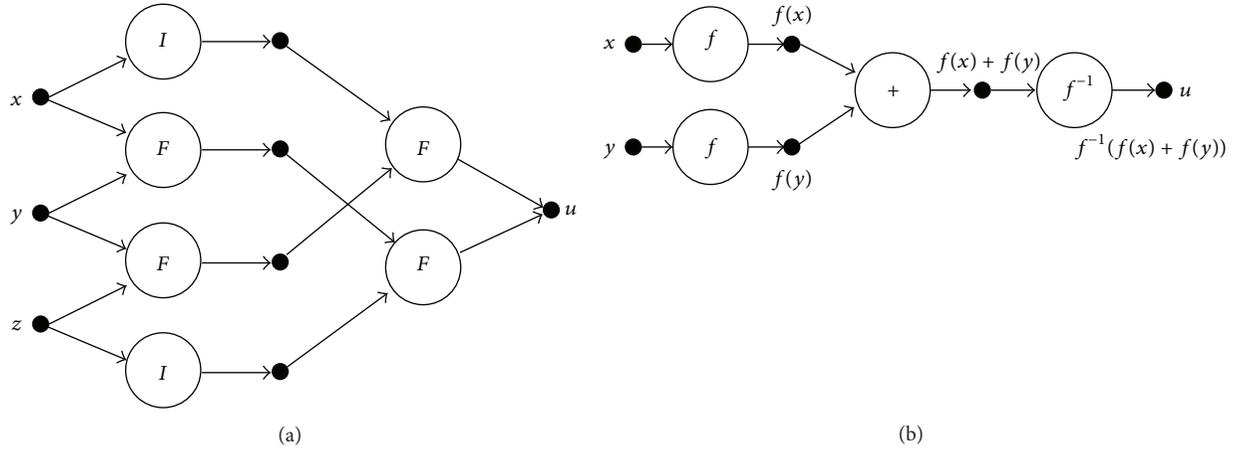


FIGURE 1: The associativity functional network: (a) original network; (b) simplified network.

set is optional and allows more than one neuron output to be connected to the same unit. In Figure 1(b), there are two intermediate layers of storing units, which are represented by small circles in black.

(c) *A Layer of Output Units.* This last layer contains the output information. In Figure 1(b), this output layer is reduced to the unit  $u = f^{-1}(f(x) + f(y))$ .

(ii) *One or More Layers of Neurons or Computing Units.* A neuron is a computing unit which evaluates a set of input values, coming from the previous layer, of input or intermediate units, and gives a set of output values to the next layer, of intermediate or output units. Neurons are represented by circles with the name of the corresponding neural function inside. For example, in Figure 1(b), we have three layers of neurons. The first one gives outputs of functions with one variable. The second layer exhibits the sum operator of its two inputs. The last layer computes the inverse of the first layer function applied to output of the previous layer.

(iii) *A Set of Directed Links.* They connect the input or intermediate layers to its adjacent layer of neurons and neurons of one layer to its adjacent intermediate layers or to the output layer. Connections are represented by arrows, indicating the information flow direction. We remark here that information flows in only one direction, from the input layer to the output layer.

All these elements together form the *network architecture* or *topology* of the functional network, which defines the functional capabilities of the network.

4.2. *Differences between Functional and Neural Networks.* In next paragraphs, we discuss the differences between functional and neural networks and the advantages of using functional networks instead of standard neural networks.

- (1) In neural networks, each neuron returns an output  $y = f(\sum w_{ik}x_k)$  that depends only on the value

$\sum w_{ik}x_k$ , where  $x_1, x_2, \dots, x_n$  are the received inputs (see Figure 2(a)). Therefore, their neural functions have only one argument. In contrast, neural functions in functional networks can have several arguments, as shown in Figure 2(b).

- (2) In neural networks, the neural functions are *univariate*: neurons can show different outputs but all of them represent the same values. In functional networks, the neural functions can be *multivariate*.
- (3) In a given functional network, the neural functions can be *different* (such as functions  $f_1, f_2$ , and  $f_3$  in Figure 2(b)), while in neural networks they are *identical*.
- (4) In neural networks, there are weights, which must be learned. These weights do not appear in functional networks, where neural functions are learned instead.
- (5) In neural networks *the neuron outputs are different*, while in functional networks *neuron outputs can be coincident*. This fact leads to a set of functional equations, which have to be solved [73, 74]. These functional equations impose strong constraints leading to a considerable reduction in the degrees of freedom of the neural functions. In most cases, this implies that neural functions can be reduced in dimension or expressed as functions of smaller dimensions.

All these features show that the functional networks exhibit more interesting possibilities than the neural networks. This implies that some problems can be solved more efficiently by using functional networks instead of neural networks.

## 5. Our Method

In this section, we describe the proposed method for solving the surface reconstruction problem indicated in Section 3.2. Firstly, a general overview of the method is presented. Then, each step of the method is discussed in detail.

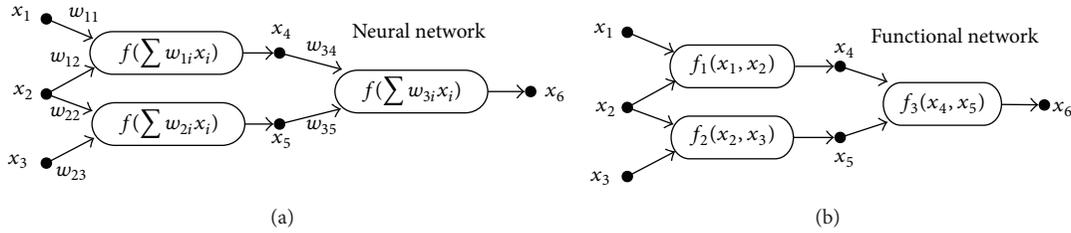


FIGURE 2: Differences between (a) neural networks and (b) functional networks.

**5.1. Overview of the Method.** The graphical workflow in Figure 3 summarizes the main steps of our method. The initial input consists of a set of irregularly sampled noisy 3D data points assumed to lie on an unknown surface. The goal is to obtain the NURBS surface that approximates these data points optimally. To this purpose, we need to solve two important subproblems: data parameterization and surface approximation. To address data parameterization, we firstly perform a principal component analysis (PCA) to obtain a parametric plane that accounts for the variability of data. Then, the data points are projected onto that parametric plane. A 2D surface parameterization is subsequently obtained by applying a Kohonen neural network to the set of projected 2D data points. Then, we apply our NURBS functional network to compute all other parameters of the NURBS surface approximating the data points for this initial parameterization. We call that surface a base surface. The next step consists of computing the fitting error for the approximating surface. Finally, the data points are projected onto the new base surface in order to yield a new (more accurate) parameterization. All previous steps are repeated iteratively until a stopping criterion is reached. Usual stopping criteria are that the fitting error becomes smaller than a given threshold value or that successive iterations of this reconstruction pipeline no longer improve current solutions.

**5.2. Data Parameterization.** The parameterization step consists of establishing the relationships among the data points in the surface parametric domain. This process is essential for a good fitting of data points. Some standard procedures are given by the uniform, chord length and centripetal parameterizations. However, these methods are only suitable for data points distributed in a uniform grid and tend to fail for unorganized, irregularly sampled data. An alternative procedure is based on the idea of projecting the data points onto an additional surface, usually called base surface, reflecting the distribution of data points and then computing a parameterization by using the projected 2D points. The simplest case of this approach consists of using a parametric plane [11], usually orthogonal to the main viewing direction of the digitizing device. A better alternative is to use a suitable 3D surface for data projection [14], usually a coarse approximation of final fitting surface, which is expected to be modified by successive improvements of this initial surface.

In our method, we combine these ideas to develop a refinement process in iterative fashion. At the initial stage, we project the data points onto a parametric plane reflecting

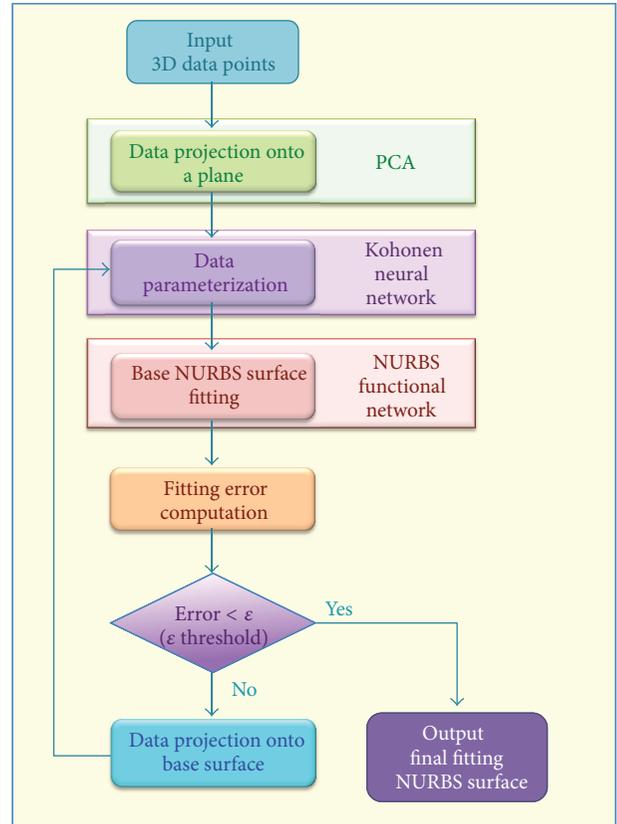


FIGURE 3: Graphical workflow of the proposed method.

the variability of data. This parametric plane is computed by using the principal component analysis (PCA), a very popular descriptive technique in data analysis and many other fields. PCA is a powerful statistical method aimed at performing dimensionality reduction by using the analysis of the correlation of data. A great advantage of this method is its nonparametric nature, meaning that it does not require any parameter tuning in order to get an output. Furthermore, the answer is unique and available regardless of the way the data has been recorded or obtained. Other main reasons for our choice are that PCA is very efficient at preserving distances between the points and each principal component has the highest variance possible under the constraint that it is uncorrelated with preceding components. In fact, the first principal component corresponds to a line passing through the multidimensional mean. It also minimizes the sum of

squares of the distances to the points from that line. Because of these properties, the parametric plane obtained by PCA is very adequate for the initial 2D projection.

Projected 2D points are then used for an initial surface parameterization by applying a Kohonen neural network (also called self-organizing map (SOM) neural network). This is a very popular artificial neural network for unsupervised learning. The interested reader is referred to the nice books in [75, 76] for a comprehensive overview about the Kohonen neural network, its fundamentals and mathematical basis, along with the most important variants and modifications and several interesting applications.

*Algorithm 1.* Kohonen neural network for 2D data parameterization.

- (1) Initialize position  $\Psi$  of all neurons  $\Phi$  with random values within the surface boundaries.
- (2) Initialize the weights  $\sigma_{i,j}$  randomly around the centroid of the input space.
- (3) Pick a new input sample  $Q_\delta$ .
- (4) Compute

$$d_j = \sqrt{\sum_{i=1}^R (\Psi_i(\delta) - \sigma_{i,j}(\delta))^2}. \quad (12)$$

- (5) Select the active neuron  $\gamma$  for which  $d_\gamma = \min_{i=1,\dots,R}(d_i)$ . The winner is the closest active neuron to the sample data  $Q_\delta$ , with parametric coordinates  $(\tilde{i}, \tilde{j})$ .
- (6) Compute the neighborhood of neuron  $\gamma$  as follows. The radius  $\rho$  is given by

$$\rho(\gamma) = \frac{R}{2e^{[\sum_{i=1}^{\delta} (1/10+i/10^4)]}}. \quad (13)$$

The bubble neighborhood of the winner neuron is defined as all neurons at positions  $(i, j)$  such that

$$|i - \tilde{i}| + |j - \tilde{j}| < \rho \left(1 - \frac{\delta}{\theta}\right), \quad (14)$$

where  $\theta$  is the run length.

- (7) Update the weight of neuron  $\gamma$  and all neurons in its neighborhood as

$$\sigma_{i,j}(\delta + 1) = \sigma_{i,j}(\delta) + \eta(\delta) (\Psi_i(\delta) - \sigma_{i,j}(\delta)) \quad (15)$$

with

$$\eta(\delta) = \zeta_\delta \frac{1}{\sqrt{2\pi e^{-\delta^2/2}}}, \quad (16)$$

where the learning rate  $\zeta_\delta$  is given by

$$\zeta_\delta = 1 - \left(1 - \zeta_0 \left(1 - \frac{\delta}{\theta}\right)\right)^{\sigma_\delta}. \quad (17)$$

- (8) Increment  $\delta$ .

- (9) Repeat the steps (3)–(8) until  $\delta$  reaches the limit value.

A very remarkable feature of the Kohonen neural network is that it incorporates a neighborhood function to preserve the topological properties of the input space [75]. This property is very useful to generate a 2D grid for parameterization, where the neurons represent the grid nodes. The neural network is trained by using the projected 2D data so that its topology eventually reflects their shape and neighborhood relations. To this purpose, each neuron contains geometrical information about the coordinates of the associated node and the topological relations with its neighbors. It is important to remark that the connections among the neurons do not change during the training stage; instead, the changes occur on the geometrical values stored in the neurons. In other words, the network performs a topological ordering of the competitive neurons such that the neighboring neurons represent clusters in the two-dimensional space. Eventually, the weights will specify cluster centers whose distribution approximates the distribution of data points and hence a suitable 2D parameterization. Because of these good properties, the Kohonen neural network has already been used in several ways for data parameterization in some previous works [49, 70, 71, 77]. The algorithm used in this paper combines some of the best features of previous methods. It is briefly summarized in Algorithm 1.

*5.3. Surface Fitting.* A major property of functional networks is their ability to reproduce the functional structure of the underlying mathematical function of the data points. In this paper, we introduce a new functional network (depicted in Figure 4) especially designed to reproduce the functional structure of NURBS surfaces: the NURBS functional network. Its workflow can be traced graphically by proceeding upwardly in Figure 4: given the surface parameter values  $u$  and  $v$  and two orders  $k$  and  $l$ , what this functional network essentially does is to compute the values of the basis functions  $N_{ik}$  at  $u$  and  $N_{jl}$  at  $v$  and then the bivariate tensor-product basis functions  $N_{ik}(u)N_{jl}(v)$ , ( $i = 0, \dots, m; j = 0, \dots, n$ ). Each bivariate basis function is then multiplied twice, firstly by the scalar weight  $w_{i,j}$  and then by such a weight and its associated vector weight  $\mathbf{P}_{i,j}$ . Summation on indices  $i$  and  $j$  is applied to both expressions to obtain the numerator and denominator of (3), to which the quotient operator is subsequently applied. Note that two different (scalar and vector) time operators are considered, to account for the multiplication by  $w_{i,j}$  and  $\mathbf{P}_{i,j}$ , respectively. Note also that those vectors  $\mathbf{P}_{i,j}$  play the role of weights of the neural functions as well, with the meaning that a functional network with  $d$ -dimensional vectors as weights can be understood as  $d$  parallel functional networks with scalar weights. Note finally that the first three layers of this functional network apply functions to either the same or independent arguments of the previous layer, so NURBS functional networks are well suited for partial bottom-up parallelization.

Now, we use the surface parameterization obtained in previous steps to compute an approximating surface to

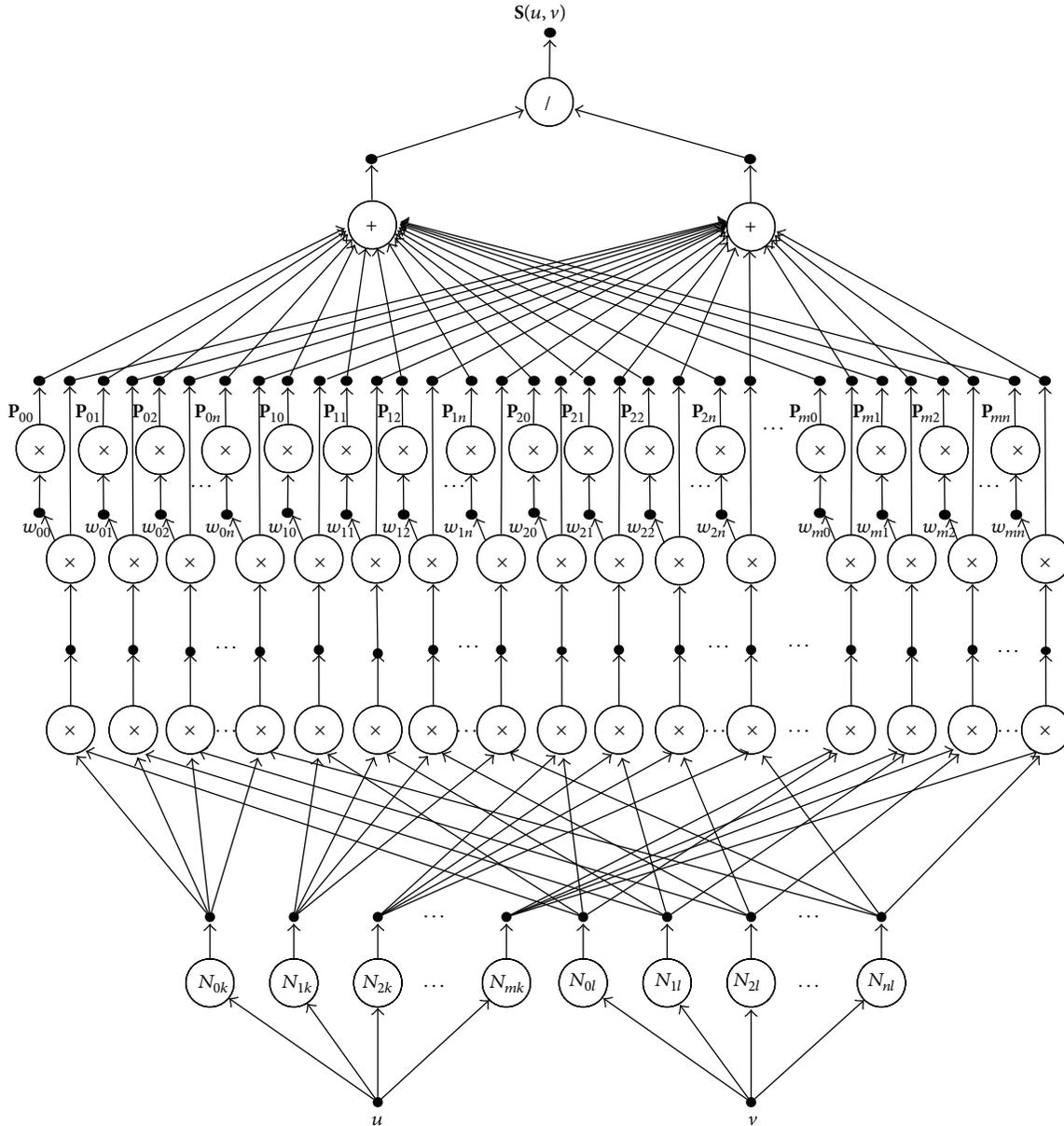


FIGURE 4: Graphical representation of the NURBS functional network.

the data points. This functional network is trained by supervised learning in which couples of input-output values  $\mathbf{D} = \{(I_i, O_i) \mid i = 1, \dots, R\}$ , corresponding to the surface parameters  $\{(u_\mu, v_\mu)\}_\mu$  and their associated data points  $\mathbf{Q}_\mu$ , are presented to the network. Unlike ANN, where the neuron functions are assumed to be fixed and known and only the weights are learned, in functional networks the functions are learned during the structural learning (which obtains the simplified network and functional structures) and estimated during the parametric learning (which consists of obtaining the optimal neuron function from a given family). In our case, the structural learning corresponds to the topology of the NURBS functional network, where the input data are the order  $(k, l)$  and the length  $(r, h)$  of the knot vectors of

the approximating surface. These values determine the number of neurons in each layer of the functional network.

The parametric learning concerns the estimation of the neuron functions. It is usually accomplished by considering linear combinations of given functional families and estimating the associated parameters from the available data. For our choice of B-spline basis functions during the structural learning, the parametric learning is required to determine the scalar and vector weights of our NURBS functional network. Vector weights are learned by using expression (8) where  $\Delta$  is a symmetric square matrix. This system is solved by using the singular value decomposition (SVD), which provides the best numerical answer for the minimization problem in (9) in those cases in which the exact solution is not possible

TABLE 1: Number of data points and input parameters of the approximating NURBS surfaces for the two examples discussed in this paper.

Example	Number of data points	Order	$(r, h)$	Iterations
Mask	11830	(4, 4)	(17, 14)	2
Umbrella	8926	(3, 3)	(6, 27)	4

(see [78] for details). Finally, the scalar weights are obtained by least-squares minimization of expression (9), where all other relevant parameters of this minimization problem have already been obtained as described in previous steps.

**5.4. Surface Refinement.** Once the approximating surface is obtained, the initial parametric plane used for data parameterization is replaced by this approximating surface, which becomes the new base surface. Data points are then projected onto this base surface by computing the nearest point on the surface to each 3D data point by following the procedure indicated in [79]. A new data parameterization and surface fitting steps are computed according to Sections 5.2 and 5.3, respectively. The resulting fitting surface becomes the new base surface and so on. In general, this procedure yields a more refined (i.e., more accurate) fitting surface to data points.

This process is repeated iteratively until a stopping criterion is reached. Usual stopping criteria are that the fitting error becomes smaller than a given threshold value or that successive iterations of this reconstruction pipeline no longer improve current solutions. In the former case, we assume that an error threshold value is provided as an input of the problem (as it usually happens in many industrial problems) and we compute the fitting error according to either (4) or (5). In the latter case, we compare the fitting error between successive iterations, and the process is stopped when no further improvement is achieved.

## 6. Illustrative Examples

Our method has been tested with several examples of different clouds of data points. To keep the paper at manageable size, we discuss here two of them. Examples in this paper are shown in Figures 5-6. For each example, two different pictures are displayed: at (a), we show the original cloud of input data points, represented as small red points; at (b), the best approximating NURBS surface, is shown as obtained with our functional-neural approach. Our input consists of sets of irregularly sampled data points (this fact can readily be seen from simple visual inspection of the point clouds at (a)), which are also affected by measurement noise of a signal-to-noise ratio of 15:1 in all examples. In this paper, a fitting error threshold value  $\epsilon = 10^{-3}$  is considered. The other relevant parameters of the approximating NURBS surfaces are reported in Table 1, where the examples are arranged in rows. For each example, the following data are arranged in columns: number of data points, order of the approximating NURBS surface, length of knot vectors, and number of

iterations required to obtain the fitting surface with a fitting error below the given threshold.

A simple visual inspection of the figures clearly shows that our method yields a very good approximating surface to data points in all cases. The low number of iterations required to obtain the fitting surface for the given threshold also confirms the good behavior of the method. From these examples and many others not reported here for the sake of brevity, we conclude that the presented method performs very well, with remarkable capability to provide a satisfactory solution to the general reconstruction problem with NURBS surfaces.

Regarding the implementation issues, all computations in this paper have been performed on a 2.9 GHz. Intel Core i7 processor with 8 GB. of RAM. The source code has been implemented by the authors in the native programming language of the popular scientific program *Matlab*, version 2010b.

## 7. Comparison with Other Approaches

In this section, we compare the presented method with other alternative approaches for surface reconstruction based on neural networks. A careful revision of the literature in the field gives six previous contributions in the field, the works in [4, 8, 49, 70, 71, 77]. This small number is a clear indication of the difficulty and originality of the present work.

Comparative results of these methods are summarized in Table 2. The different methods are arranged in rows and sorted by year of publication. For each reported method, the columns give a brief description about its main features. Columns 2, 4, and 6 give a binary answer to three different questions: whether or not the indicated method provides algorithms for the subproblems of data parameterization, surface fitting, and support for NURBS surfaces, respectively. Answer *true* is marked with a check ( $\checkmark$ ), otherwise with symbol ( $\times$ ). Wherever a positive answer is found, a short description about the specific techniques incorporated in that method for the reported subproblem is given in columns 3 and 5, respectively. Note that all methods except [4, 8] address the data parameterization subproblem with a Kohonen neural network, and all methods except [70, 77] provide a procedure to compute the approximating surface, either as a Bézier surface [8], a B-spline surface [4, 71], or a polygonal mesh [49]. The works in [70, 77] assume that any traditional surface fitting technique will be used for this particular subproblem.

The most important difference of this method with respect to alternative approaches is that previous methods do not provide support for NURBS surfaces (see column 6 in Table 2). On the contrary, NURBS surfaces are fully supported in our method. In fact, the examples shown in the paper are more difficult to reconstruct through simple polynomial surfaces and require a larger number of parameters. In clear contrast, our method requires only a small number of parameters. Furthermore, the solution is reached with a small number of iterations. To the best of our knowledge, this is the first neural-based approach providing full support for all

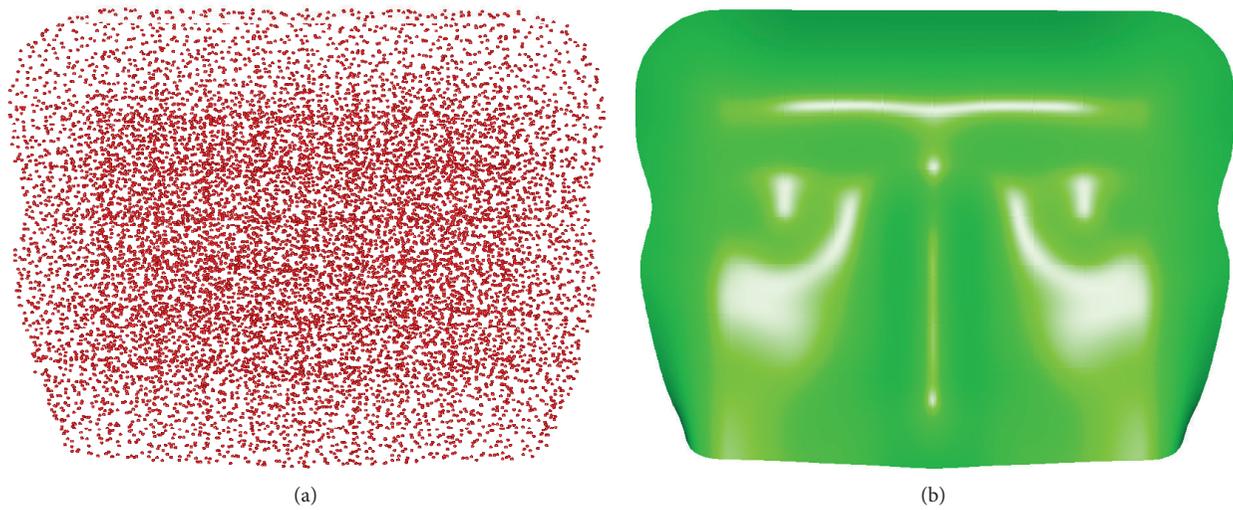


FIGURE 5: Mask example: (a) original data points; (b) fitting surface.

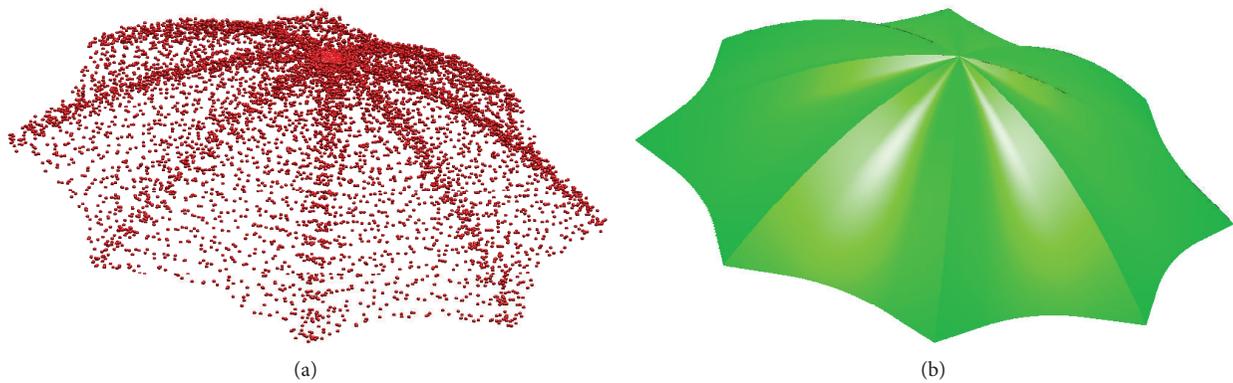


FIGURE 6: Umbrella example: (a) original data points; (b) fitting surface.

steps of the general surface reconstruction problem by using NURBS surfaces.

## 8. Conclusions and Future Work

This paper proposes a new hybrid functional-neural approach to solve the surface reconstruction problem. In our approach, we combine two powerful artificial intelligence paradigms: the popular Kohonen neural network to address the data parameterization problem and a new functional network, called NURBS functional network, to reproduce the functional structure of the NURBS surfaces. These neural and functional networks are then applied iteratively for further surface refinement. The hybridization of these two networks provides us with a powerful computational

approach to solve the surface reconstruction problem. The performance of our approach has been tested by its application to two illustrative examples. Our results show that the method performs very well, being able to reconstruct the approximating surface of the given set of data points with a high degree of accuracy and a low number of iterations.

The main limitation of this approach is that it requires some initial input such as the order of the approximating NURBS surface and the length of knot vectors, which are strongly dependent on the particular set of data points. Consequently, their optimal values might be difficult to choose for end users, thus preventing the method for automatic, human-independent reconstruction. This limitation opens the door for future research in the area in order to develop efficient algorithms for automatic determination of those optimal

TABLE 2: Comparison of the proposed method with other alternative methods based on neural networks for the surface reconstruction problem.

Author, year, and reference	Data parameterization	Method used	Surface fitting	Method used	NURBS supported?
Hoffmann and Várady (1998) [70]	✓	Kohonen neural network	×		×
Yu (1999) [49]	✓	Kohonen neural network	✓	Polygonal mesh (edge swap)	×
Hoffmann (1999) [77]	✓	Modified Kohonen neural network	×		×
Barhak and Fischer (2001) [71]	✓	(i) Kohonen neural network (ii) Partial differential equations (PDE)	✓	(i) Gradient descent algorithm (GDA) (ii) Random surface error correction (RSEC)	×
Iglesias et al. (2004) [4]	×		✓	Tensor-product functional network	×
Gálvez et al. (2007) [8]	✓	Genetic algorithms	✓	Functional network	×
This method (2013)	✓	Kohonen neural network	✓	NURBS functional network	✓

values. Another important limitation of the method occurs when the data points cannot be unambiguously projected onto the base surface. This problem can arise with closed surfaces often represented in implicit form. In those cases, the PCA method might fail to extract the real tendency of data. We are currently working on new strategies to overcome this problem. Future work also includes the extension of this approach to other kinds of approximating surfaces as well as the possible application of this methodology to some interesting industrial problems.

### Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper. Any commercial identity mentioned in this paper is cited solely for scientific purposes.

### Acknowledgments

This research has been kindly supported by the Computer Science National Program of the Spanish Ministry of Economy and Competitiveness, Project ref. no. TIN2012-30768, Toho University (Funabashi, Japan), and the University of Cantabria (Santander, Spain). The authors are particularly grateful to the Department of Information Science of Toho University for all the facilities given to carry out this work. Special thanks are owed to the editor and the four anonymous reviewers for their useful comments and suggestions that allowed us to improve the final version of this paper.

### References

- [1] T. Várady, R. R. Martin, and J. Cox, "Reverse engineering of geometric models—an introduction," *CAD Computer Aided Design*, vol. 29, no. 4, pp. 255–268, 1997.
- [2] T. Várady and R. Martin, "Reverse engineering," in *Handbook of Computer Aided Geometric Design*, G. Farin, J. Hoschek, and M. Kim, Eds., pp. 651–681, Elsevier Science, 2002.
- [3] R. E. Barnhill, *Geometric Processing for Design and Manufacturing*, SIAM, Philadelphia, Pa, USA, 1992.
- [4] A. Iglesias, G. Echevarría, and A. Gálvez, "Functional networks for B-spline surface reconstruction," *Future Generation Computer Systems*, vol. 20, no. 8, pp. 1337–1353, 2004.
- [5] A. Iglesias and A. Gálvez, "A new artificial intelligence paradigm for computer aided geometric design," in *Artificial Intelligence and Symbolic Computation*, vol. 1930 of *Lectures Notes in Artificial Intelligence*, pp. 200–213, 2001.
- [6] N. M. Patrikalakis and T. Maekawa, *Shape Interrogation for Computer Aided Design and Manufacturing*, Springer, Heidelberg, Germany, 2002.
- [7] H. Pottmann, S. Leopoldseder, M. Hofer, T. Steiner, and W. Wang, "Industrial geometry: recent advances and applications in CAD," *CAD Computer Aided Design*, vol. 37, no. 7, pp. 751–766, 2005.
- [8] A. Gálvez, A. Iglesias, A. Cobo, J. Puig-Pey, and J. Espinola, "Bézier curve and surface fitting of 3D point clouds through genetic algorithms, functional networks and least-squares approximation," in *Computational Science and Its Applications—ICCSA 2007*, vol. 4706 of *Lectures Notes in Computer Science*, pp. 680–693, 2007.
- [9] G. E. Farin, *Curves and Surfaces for Computer-Aided Geometric Design*, Morgan Kaufmann, San Francisco, Calif, USA, 5th edition, 2001.
- [10] A. Gálvez and A. Iglesias, "Efficient particle swarm optimization approach for data fitting with free knot B-splines," *CAD Computer Aided Design*, vol. 43, no. 12, pp. 1683–1692, 2011.
- [11] J. Hoschek and D. Lasser, *Fundamentals of Computer Aided Geometric Design*, A. K. Peters, 1993.
- [12] J. Ling and S. Li, "Fitting B-spline curves by least squares support vector machines," in *Proceedings of the International Conference on Neural Networks & Brain Proceedings (ICNNB '05)*, pp. 905–909, IEEE Press, Beijing, China, October 2005.
- [13] T. C. M. Lee, "On algorithms for ordinary least squares regression spline fitting: a comparative study," *Journal of Statistical Computation and Simulation*, vol. 72, no. 8, pp. 647–663, 2002.
- [14] W. Y. Ma and J. P. Kruth, "Parameterization of randomly measured points for least squares fitting of B-spline curves and surfaces," *Computer-Aided Design*, vol. 27, no. 9, pp. 663–675, 1995.

- [15] W. P. Wang, H. Pottmann, and Y. Liu, "Fitting B-spline curves to point clouds by curvature-based squared distance minimization," *ACM Transactions on Graphics*, vol. 25, no. 2, pp. 214–238, 2006.
- [16] H. P. Yang, W. P. Wang, and J. G. Sun, "Control point adjustment for B-spline curve approximation," *CAD Computer Aided Design*, vol. 36, no. 7, pp. 639–652, 2004.
- [17] P. Dierckx, *Curve and Surface Fitting with Splines*, Oxford University Press, Oxford, UK, 1993.
- [18] H. Park and J.-H. Lee, "B-spline curve fitting based on adaptive curve refinement using dominant points," *CAD Computer Aided Design*, vol. 39, no. 6, pp. 439–451, 2007.
- [19] L. A. Piegl and W. Tiller, "Least-squares B-spline curve approximation with arbitrary end derivatives," *Engineering with Computers*, vol. 16, no. 2, pp. 109–116, 2000.
- [20] D. L. B. Jupp, "Approximation to data by splines with free knots," *SIAM Journal on Numerical Analysis*, vol. 15, no. 2, pp. 328–343, 1978.
- [21] J. R. Rice, *The Approximation of Functions*, vol. 2, Addison-Wesley, Reading, Mass, USA, 1969.
- [22] J. Hertz, A. Krogh, and R. G. Palmer, *Introduction to the Theory of Neural Computation*, Addison-Wesley, Reading, Mass, USA, 1991.
- [23] M. G. Cox, "Algorithms for spline curves and surfaces," in *Fundamental Developments of Computer-Aided Geometric Design*, L. Piegl, Ed., pp. 51–76, Academic Press, London, UK, 1993.
- [24] R. H. Franke and L. L. Schumaker, "A bibliography of multivariate approximation," in *Topics in Multivariate Approximation*, C. K. Chui, L. L. Schumaker, and F. I. Utreras, Eds., Academic Press, New York, NY, USA, 1986.
- [25] N. R. Draper and H. Smith, *Applied Regression Analysis*, Wiley-Interscience, 3rd edition, 1998.
- [26] D. R. Forsey and R. H. Bartels, "Surface fitting with hierarchical splines," *ACM Transactions on Graphics*, vol. 14, pp. 134–161, 1995.
- [27] H. Pottmann, S. Leopoldseeder, and M. Hofer, "Approximation with active bspline curves and surfaces," in *Proceedings of the 10th Pacific Conference on Computer Graphics and Applications*, pp. 8–25, IEEE Computer Society Press, 2002.
- [28] M. Eck and H. Hoppe, "Automatic reconstruction of B-spline surfaces of arbitrary topological type," in *Proceedings of the Computer Graphics Conference (SIGGRAPH '96)*, pp. 325–334, August 1996.
- [29] L. Piegl and W. Tiller, *The NURBS Book*, Springer, Berlin, Germany, 1997.
- [30] M. Prasad, A. Zisserman, and A. Fitzgibbon, "Single view reconstruction of curved surfaces," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '06)*, vol. 2, pp. 1345–1354, IEEE Computer Society Press, Los Alamitos, Calif, USA, June 2006.
- [31] M. Levoy, K. Pulli, B. Curless et al., "The digital Michelangelo project: 3D scanning of large statues," in *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '00)*, pp. 131–144, New Orleans, La, USA, July 2000.
- [32] M. C. Leu, X. Peng, and W. Zhang, "Surface reconstruction for interactive modeling of freeform solids by virtual sculpting," *CIRP Annals—Manufacturing Technology*, vol. 54, no. 1, pp. 131–134, 2005.
- [33] C. V. Alvino and A. J. Yezzi Jr., "Tomographic reconstruction of piecewise smooth images," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '04)*, vol. 1, pp. 1576–1581, Los Alamitos, Calif, USA, July 2004.
- [34] J. Rossignac and B. Borrel, "Multi-resolution 3D approximations for rendering complex scenes," in *Geometric Modeling in Computer Graphics*, pp. 455–465, Springer, 1993.
- [35] C. L. Bajaj, F. Bernardini, and G. Xu, "Automatic reconstruction of surfaces and scalar fields from 3D scans," in *Proceedings of the 22nd Annual ACM Conference on Computer Graphics and Interactive Techniques*, pp. 109–118, August 1995.
- [36] C. L. Bajaj, E. J. Coyle, and K.-N. Lin, "Arbitrary topology shape reconstruction from planar cross sections," *Graphical Models and Image Processing*, vol. 58, no. 6, pp. 524–543, 1996.
- [37] M. Jones and M. Chen, "A new approach to the construction of surfaces from contour data," *Computer Graphics Forum*, vol. 13, no. 3, pp. 75–84, 1994.
- [38] D. Meyers, S. Skinnwer, and K. Sloan, "Surfaces from contours," *ACM Transactions on Graphics*, vol. 11, no. 3, pp. 228–258, 1992.
- [39] H. Park and K. Kim, "Smooth surface approximation to serial cross-sections," *CAD Computer Aided Design*, vol. 28, no. 12, pp. 995–1005, 1996.
- [40] W. J. Gordon, "Spline-blended surface interpolation through curve networks," *Journal of Mathematics and Mechanics*, vol. 18, no. 10, pp. 931–952, 1969.
- [41] H. Fuchs, Z. M. Kedem, and S. P. Useton, "Optimal surface reconstruction from planar contours," *Communications of the ACM*, vol. 20, no. 10, pp. 693–702, 1977.
- [42] T. Maekawa and K. H. Ko, "Surface construction by fitting unorganized curves," *Graphical Models*, vol. 64, no. 5, pp. 316–332, 2002.
- [43] V. Savchenko, A. Pasko, O. Okunev, and T. Kunii, "Function representation of solids reconstructed from scattered surface points and contours," *Computer Graphics Forum*, vol. 14, no. 4, pp. 181–188, 1995.
- [44] G. Echevarría, A. Iglesias, and A. Gálvez, "Extending neural networks for B-spline surface reconstruction," in *Computational Science—ICCS*, vol. 2330 of *Lectures Notes in Computer Science*, pp. 305–314, 2002.
- [45] P. Gu and X. Yan, "Neural network approach to the reconstruction of freeform surfaces for reverse engineering," *Computer-Aided Design*, vol. 27, no. 1, pp. 59–64, 1995.
- [46] B. Guo, "Surface reconstruction: from points to splines," *CAD Computer Aided Design*, vol. 29, no. 4, pp. 269–277, 1997.
- [47] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle, "Surface reconstruction from unorganized points," *Computer Graphics (ACM)*, vol. 26, no. 2, pp. 71–78, 1992.
- [48] C. T. Lim, G. M. Turkiyyah, M. A. Ganter, and D. W. Storti, "Implicit reconstruction of solids from cloud point sets," in *Proceedings of the 3rd Symposium on Solid Modeling and Applications*, pp. 393–402, May 1995.
- [49] Y. Yu, "Surface reconstruction from unorganized points using self-organizing neural networks," in *Proceedings of IEEE Visualization '99*, pp. 61–64, 1999.
- [50] M. S. Floater, "Parametrization and smooth approximation of surface triangulations," *Computer Aided Geometric Design*, vol. 14, no. 3, pp. 231–250, 1997.
- [51] Č. Oblonšek and N. Guid, "A fast surface-based procedure for object reconstruction from 3D scattered points," *Computer Vision and Image Understanding*, vol. 69, no. 2, pp. 185–195, 1998.
- [52] R. M. Bolle and B. C. Vemuri, "On three-dimensional surface reconstruction methods," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 1, pp. 1–13, 1991.

- [53] F. J. M. Schmitt, B. A. Barsky, and W.-H. Du, "An adaptive subdivision method for surface fitting from sampled data," *Computer Graphics (ACM)*, vol. 20, no. 4, pp. 179–188, 1986.
- [54] T. A. Foley, "Interpolation of scattered data on a spherical domain," in *Algorithms for Approximation II*, J. C. Mason and M. G. Cox, Eds., pp. 303–310, Chapman and Hall, London, UK, 1990.
- [55] S. Sclaroff and A. Pentland, "Generalized implicit functions for computer graphics," *ACM Siggraph Computer Graphics*, vol. 25, no. 4, pp. 247–250, 1991.
- [56] V. Pratt, "Direct least-squares fitting of algebraic surfaces," *Computer Graphics (ACM)*, vol. 21, no. 4, pp. 145–152, 1987.
- [57] A. Gálvez, A. Cobo, J. Puig-Pey, and A. Iglesias, "Particle swarm optimization for Bézier surface reconstruction," in *Computational Science—ICCS 2008*, vol. 5102 of *Lectures Notes in Computer Science*, pp. 116–125, 2008.
- [58] A. Gálvez and A. Iglesias, "Particle swarm optimization for non-uniform rational B-spline surface reconstruction from clouds of 3D data points," *Information Sciences*, vol. 192, pp. 174–192, 2012.
- [59] A. Gálvez, A. Iglesias, and J. Puig-Pey, "Iterative two-step genetic-algorithm-based method for efficient polynomial B-spline surface reconstruction," *Information Sciences*, vol. 182, no. 1, pp. 56–76, 2012.
- [60] M. Sarfraz and S. A. Raza, "Capturing outline of fonts using genetic algorithms and splines," in *Proceedings of the 5th International Conference on Information Visualization (IV' 01)*, pp. 738–743, IEEE Computer Society Press, 2001.
- [61] F. Yoshimoto, M. Moriyama, and T. Harada, "Automatic knot adjustment by a genetic algorithm for data fitting with a spline," in *Proceedings of the International Conference on Shape Modeling*, pp. 162–169, IEEE Computer Society Press, 1999.
- [62] F. Yoshimoto, T. Harada, and Y. Yoshimoto, "Data fitting with a spline using a real-coded genetic algorithm," *CAD Computer Aided Design*, vol. 35, no. 8, pp. 751–760, 2003.
- [63] A. Gálvez, A. Iglesias, and A. Andreina, "Discrete Bézier curve fitting with artificial immune systems," in *Intelligent Computer Graphics 2012*, vol. 441 of *Studies in Computational Intelligence*, pp. 59–75, 2013.
- [64] E. Ülker and A. Arslan, "Automatic knot adjustment using an artificial immune system for B-spline curve approximation," *Information Sciences*, vol. 179, no. 10, pp. 1483–1494, 2009.
- [65] X. Zhao, C. Zhang, B. Yang, and P. Li, "Adaptive knot placement using a GMM-based continuous optimization algorithm in B-spline curve approximation," *CAD Computer Aided Design*, vol. 43, no. 6, pp. 598–604, 2011.
- [66] A. Gálvez and A. Iglesias, "Firey algorithm for polynomial Bézier surface parameterization," *Journal of Applied Mathematics*, vol. 2013, Article ID 237984, 9 pages, 2013.
- [67] A. Gálvez and A. Iglesias, "Firey algorithm for explicit B-spline curve fitting to data points," *Mathematical Problems in Engineering*, vol. 2013, Article ID 528215, 12 pages, 2013.
- [68] A. Gálvez and A. Iglesias, "A new iterative mutually-coupled hybrid GA-PSO approach for curve fitting in manufacturing," *Applied Soft Computing*, vol. 13, no. 3, pp. 1491–1504, 2013.
- [69] A. Gálvez and A. Iglesias, "From nonlinear optimization to convex optimization through firey algorithm and indirect approach with applications to CAD/CAM," *The Scientific World Journal*, vol. 2013, Article ID 283919, 10 pages, 2013.
- [70] M. Hoffmann and L. Várady, "Free-form surfaces for scattered data by neural networks," *Journal for Geometry and Graphics*, vol. 2, no. 1, pp. 1–6, 1998.
- [71] J. Barhak and A. Fischer, "Parameterization and reconstruction from 3D scattered points based on neural network and PDE techniques," *IEEE Transactions on Visualization and Computer Graphics*, vol. 7, no. 1, pp. 1–16, 2001.
- [72] E. Castillo, "Functional networks," *Neural Processing Letters*, vol. 7, no. 3, pp. 151–159, 1998.
- [73] E. Castillo, A. Iglesias, and R. Ruiz-Cobo, *Functional Equations in Applied Sciences*, Elsevier Science, Amsterdam, The Netherlands, 2005.
- [74] E. Castillo and A. Iglesias, "Some characterizations of families of surfaces using functional equations," *ACM Transactions on Graphics*, vol. 16, no. 3, pp. 296–318, 1997.
- [75] T. Kohonen, *Self-Organization and Associative Memory*, vol. 8 of *Springer Series in Information Sciences*, Springer, New York, NY, USA, 3rd edition, 1989.
- [76] T. Kohonen, *Self-Organizing Maps*, vol. 30 of *Springer Series in Information Sciences*, Springer, Berlin, Germany, 3rd edition, 2001.
- [77] M. Hoffmann, "Modified Kohonen neural network for surface reconstruction," *Publicationes Mathematicae*, vol. 54, pp. 857–864, 1999.
- [78] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes*, Cambridge University Press, Cambridge, UK, 2nd edition, 1986.
- [79] J. Puig-Pey, A. Gálvez, A. Iglesias, J. Rodríguez, P. Corcuera, and F. Gutiérrez, "Some applications of scalar and vector fields to geometric processing of surfaces," *Computers & Graphics*, vol. 29, no. 5, pp. 723–729, 2005.

## Research Article

# Solving the Balanced Academic Curriculum Problem Using the ACO Metaheuristic

**José-Miguel Rubio,<sup>1,2</sup> Wenceslao Palma,<sup>1</sup> Nivaldo Rodríguez,<sup>1</sup> Ricardo Soto,<sup>1,3</sup>  
Broderick Crawford,<sup>1,4</sup> Fernando Paredes,<sup>5</sup> and Guillermo Cabrera<sup>1,6</sup>**

<sup>1</sup> *Escuela de Ingeniería Informática, Pontificia Universidad Católica de Valparaíso, 2362807 Valparaíso, Chile*

<sup>2</sup> *Departamento de Computación e Informática, Universidad de Playa Ancha, 2360002 Valparaíso, Chile*

<sup>3</sup> *Universidad Autónoma de Chile, 7500138 Santiago, Chile*

<sup>4</sup> *Universidad Finis Terrae, 7501015 Santiago, Chile*

<sup>5</sup> *Escuela de Ingeniería Industrial, Universidad Diego Portales, 8370109 Santiago, Chile*

<sup>6</sup> *Department of Engineering Science, University of Auckland, Auckland 1020, New Zealand*

Correspondence should be addressed to José-Miguel Rubio; [jose.rubio.l@ucv.cl](mailto:jose.rubio.l@ucv.cl)

Received 4 May 2013; Revised 3 November 2013; Accepted 4 November 2013

Academic Editor: Yudong Zhang

Copyright © 2013 José-Miguel Rubio et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The balanced academic curriculum problem consists in the assignation of courses to academic periods satisfying all the load limits and prerequisite constraints. In this paper, we present the design of a solution to the balanced academic curriculum problem based on the ACO metaheuristic, in particular via the Best-Worst Ant System. We provide an experimental evaluation which illustrates the effectiveness of the proposed approach on a set of classic benchmarks as well as on real instances.

## 1. Introduction

The balanced academic curriculum problem (BACP) consists in assigning courses to teaching periods satisfying prerequisites and balancing students' load in terms of credits and number of courses [1]. The BACP planning horizon is divided into academic years, and each academic year is divided into terms. Each term of a year is a teaching period in which courses can take place. The problem consists in finding an assignment of courses to periods that satisfies certain load limits and prerequisites.

The BACP is strongly NP-complete as shown in [1]. Moreover, BACP is an interesting problem because it is at the boundary of several classes of problems such as bin packing, scheduling, and balancing [2]. For instance, prerequisite constraints make BACP look like a scheduling problem. A unique resource (the student) is balanced considering unit-time (periods) activities (courses) and temporal constraints (prerequisites).

The original formulation of BACP was proposed in [3] and was included as a benchmark, called prob030, in CSPLib.

The BACP has been tackled using constraint programming (CP) [2], integer linear programming (ILP) [4, 5], hybrid techniques based on genetic algorithms and constraint propagation [6, 7], integer programming (IP), and hybrid local search methods [1]. In [2], a CP approach is proposed where an ad hoc branching heuristic is used in order to guide the search towards balanced solutions improving the first-fail heuristic. In [4], ILP and CP are integrated in order to bound and guide the search and to reduce the search space, respectively. Moreover, the experimental evaluation shows that this integration decreases the run-time on many instances. Approaches based on ILP and CP techniques are proposed separately in [5] obtaining successful results on real instances. The benefits of hybridization have been shown in [6, 7] where a hybrid framework including genetic algorithms and constraint propagation is proposed and tested successfully. In [1], IP can reach good results in some cases using the basic formulation combined with a problem decomposition and relaxation. In the other cases, local search solvers are designed using the concept of generalized local Search Machines where different machines that combine Hill

Climbing, Simulated Annealing, and Dynamic Tabu Search are used in order to improve the results returned by IP.

In this paper we solve the BACP using the ant colony optimization (ACO) metaheuristic [8] model called Best-Worst Ant System (BWAS) [9, 10]. To the best of our knowledge this work is the first one tackling the BACP via the BWAS. BWAS integrates concepts from the evolutionary computing field allowing a good balance between exploration and exploitation of the search space provided by a mutation of the pheromone matrix and a pheromone updating mechanism, respectively. Additionally, a restart process avoids the algorithm getting stuck and performing unnecessary iterations.

Recent work shows that the balancing criterion is interesting not only for university timetabling but also for other scheduling applications, such as the allocation of workload to employees or, in that specific example, to nurses. We hope that the models and the techniques discussed in this work may contribute to further research in this context.

The rest of this paper is organized as follows. In Section 2, we introduce the BACP. In Section 3, we describe the design of our solution. In Section 4, we provide an experimental evaluation of our solution on benchmark and real instances. Section 5 concludes the paper and gives some direction for further work.

## 2. The BACP Formulation

In this section we present the BACP formulation given in [1]. The curriculum consists of entities which must satisfy academic and administrative constraints.

- (i) *Courses*. An academic curriculum is denoted by a set of courses and a set of prerequisites related to these. Let  $C$  be the set of courses to be taught during the planning horizon of a university degree. Each course  $c \in C$  gives a number of credits  $cr \in \mathbb{Z}^+$ .
- (ii) *Periods*. The planning horizon is divided into academic years, and each academic year is divided into terms. Each term is a teaching period in which courses can take place. Let  $P$  be the set of teaching periods, uniquely identified by the corresponding year and term. For example, a three-year degree organized in four terms per year has 12 periods, that is,  $P = \{1 \dots 12\}$ , and the first terms of each year are  $\{1, 5, 9\}$ .
- (iii) *Load Limits*. A minimum and a maximum number of courses, denoted by  $m$  and  $M$ , respectively, can be assigned to each term.
- (iv) *Prerequisites*. Based on their content, some courses have prerequisites, that is, a set of courses that the students must attend earlier. Prerequisites are formalized by a precedence graph, that is, a directed acyclic graph  $D = (V, A)$ . Each vertex  $i \in V$  represents a course, and each arc  $(i, j) \in A$  represents a precedence relation, stating that the course  $i$  is a prerequisite of course  $j$ . If course  $i$  is a prerequisite of course  $j$ , then it has to

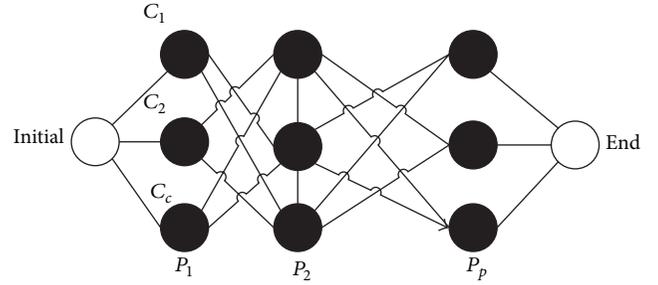


FIGURE 1: Construction graph.

be assigned to a teaching period that strictly precedes the one assigned to course  $j$ .

- (v) *Equal Distribution of Load*. The distribution of credits among the teaching periods must be balanced. Ideally, each term should have the same number of credits.

## 3. Tackling the BACP Using the ACO Metaheuristic

In this section, we describe our approach to tackle the BACP using an ACO model called Best-Worst Ant System (BWAS) [10]. BWAS has three main components.

- (i) *A Best-Worst Pheromone Trail Update Rule*. This rule reinforces the edges contained in good solutions and penalizes all the edges contained in the current worst solution.
- (ii) *Pheromone Trail Mutation*. The pheromone trails are mutated, with a certain probability, in order to introduce diversity in the search process. Pheromone trails are mutated by adding or subtracting the same amount of pheromone.
- (iii) *Restart of the Search Process*. When pheromone trails related to the edges belonging to the best solutions are very high and the remaining ones are close to zero (stagnation phase), the pheromone matrix is set to the initial pheromone value.

In the rest of this section, we describe the graph that represents the assignment of courses to periods and we detail the algorithm of our proposal based on the description of the aforementioned main issues.

**3.1. Construction Graph.** The graph that represents the assignment of the courses to determined periods follows a particular behaviour compared with a TSP. In this case, the graph must have  $|C| \times |P|$  nodes, where  $C$  is the total amount of courses and  $P$  is the total amount of periods.

Figure 1 shows the graph used to represent the assignment of courses to teaching periods. For each course the ant must choose a period from the initial node to the end node. Thus, horizontally, the ant can pass in one node only. This is because one course is assigned to one period, but a period can be assigned to many courses. However, the representation of the graph in the implementation is given by a vector

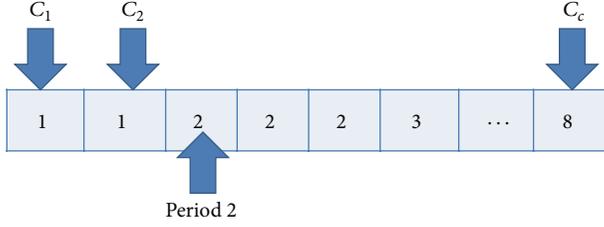


FIGURE 2: Representation of the graph using a vector.

representation as we show in Figure 2. The length of the vector is equal to the number of courses of the curriculum and each position represents the value of the period assigned to that course. Following the example of Figure 2, period 1 is assigned to course 1, period 1 is assigned to course 2, and so on until the course  $C$ .

To determine the period assigned to each course, a transition rule, which guides the ants in the construction of their solutions, is used. It has two elements: the pheromone trace and the heuristic information. The transition rule is the following:

$$P(c, p_i) = \frac{[\tau_{(c, p_i)}]^\alpha [\eta_{(c, p_i)}]^\beta}{\sum_{u \in P} ([\tau_{(c, p_u)}]^\alpha [\eta_{(c, p_u)}]^\beta)}. \quad (1)$$

**3.2. The Bwas Algorithm.** In this section we give the details of the proposed algorithm to tackle the BACP using Bwas. Thus, we focus on how to represent courses and periods, how to update/mutate pheromone trails, and how to restart the search process when it gets stuck. These are the main issues to solve the BACP using Bwas. The details are in Algorithm 1.

The ants deposit pheromone traces when they build a solution to help guide the rest of the colony. In our approach, the pheromone matrix stores the load of pheromone in a certain node instead of showing the quantity of pheromones associated to an edge between two nodes. Let  $T[1 \cdots C, 1 \cdots P]$  be the pheromone matrix, where  $\tau_{ij}$  ( $1 \leq i \leq C$ ,  $1 \leq j \leq P$ ) is an element of  $T$  representing the pheromone trace deposited by an ant when the course  $i$  is assigned to period  $j$ . To initialize the pheromone matrix, an initial solution, which only satisfies the prerequisite constraint, is calculated. Once this initial solution is obtained, we calculate the pheromone trail using the quality  $Q(S_k)$  of the obtained solution. The quality of a solution is calculated with respect to maximum load assigned to the academic periods using the following expression:

$$Q(\text{Sol}) = \frac{1}{\text{Max}\{c_1, c_2, \dots, c_p\}}. \quad (2)$$

One of the main differences between natural and artificial ants is that the artificial ants use the pheromone trails and also information named heuristic information, denoted by  $\eta(c, p_i)$ , which measures the preference of moving from one node to another [1]. In static problems this value does not change but in the BACP it is required to be changed in

```

(1)  $\tau_o \leftarrow \text{InitialSolution}(I)$ 
(2) initializePheromoneTrails( $\tau_o$ )
(3) while (term criterion not satisfied) do
(4)   for  $i = 1$  to  $k$  do
(5)     for  $j = 1$  to numCourses do
(6)       period  $\leftarrow$  transitionRule( $j$ )
(7)       assignCourse( $j$ , period)
(8)     end for
(9)     deposition( $Q(S_i)$ )
(10)    end for
(11)    evaporation()
(12)    currentBest  $\leftarrow$  selectBestSolutionIt( $A_k$ )
(13)    localSearch(currentBest)
(14)    if (best(currentBest, globalBest)) then
(15)      globalBest  $\leftarrow$  currentBest
(16)    end if
(17)    depositBestAnt(globalBest)
(18)    currentWorst  $\leftarrow$  selectWorstSolution( $A_k$ )
(19)    evaporateWorstAnt(currentWorst)
(20)    mutation()
(21)    if (stagnationcondition) then
(22)      reinitializePheromoneTrails()
(23)    end if
(24)  end while

```

ALGORITHM 1: Bwas\_bacp (instance  $I$ : a BACP instance).

order to give more information to the transition rule. Thus, we defined a function that allows evaluating the amount of violations generated when an assignment is made:

$$\eta(c, p_i) = \frac{1}{1 + v_d(c, p_i) - v_a(c, p_i)}, \quad (3)$$

where  $v_d(c, p_i)$  determines the number of violations generated when the course  $c$  is assigned to period  $p_i$  and  $v_a(c, p_i)$  determines the number of violations before this assignment. Thus, this equation can determine how many conflicts are generated because of the assignment. If there are new violations, the heuristic information and the probability of choosing that period become smaller. This expression has a singularity when  $v_d = 1$  and  $v_a = 2$ . This situation shows that the number of violated constraints decreases and the node should be more preferable. Thus, we set  $v_a = 1$ .

There are two ways to update the pheromone trails in the pheromone matrix [10].

(i) *Deposition.* It depends on the quality of the generated solution. This update process (see (4)) is performed by all the ants. Additionally the ant which generates the best solution performs this action in order to reinforce the pheromone trail:

$$\tau_{ij} \leftarrow \tau_{ij} + f(Q(S_k)). \quad (4)$$

(ii) *Evaporation.* After the construction of the solution, evaporation (see (5)) is performed in all the nodes

of the pheromone matrix using the evaporation constant  $\rho$ . Additionally, in order to penalize, the nodes belonging to the worst solution are evaporated:

$$\tau_{ij} \leftarrow \tau_{ij} \cdot (1 - \rho). \quad (5)$$

The uniqueness of the BWAS is that which includes three daemon actions, being the Best-Worst update rule responsible for the two first ones. (1) The best ant is selected and an extra deposit is made according to the quality of the solution. (2) The worst ant is selected and evaporation is made on the nodes that have been traversed by this ant; note that evaporation is made only if the best ant has not traversed those nodes. (3) The third action is the mutation, which brings diversity to the search space, adding or subtracting pheromone trace to certain nodes:

$$\text{mut}(it, \tau_{\text{threshold}}) = \frac{it - it_r}{\text{Nit} - it_r} \cdot \sigma \cdot \tau_{\text{threshold}}, \quad (6)$$

where  $it$  is the actual iteration,  $it_r$  is the last iteration when a reset due to stagnation was performed,  $\text{Nit}$  is the total number of iterations, and  $\sigma$  is the power of mutation [10]. The threshold is calculated using the pheromone deposited in the best solution divided by the number of courses. With this mutation operator, we can add or subtract pheromone trace to the pheromone matrix. Finally, in the case that the search stagnates, a reset of the pheromone trails is performed in order to go back to the last global optimum; thus the search starts from a good solution. In this algorithm we perform a reset if there is no change after 20% of the total number of iterations. In addition, a local search is performed using the best ant of the iteration. This subprocess consists in selecting all the courses of the period with a bigger academic load. Then, we select the period with the smaller academic load and we try to reassign one course of the bigger period to the smaller period. To choose this course, we first evaluate that the course does not generate new violations. Finally, we evaluate this new solution and the last one and we keep the better one as actual best.

## 4. Experimental Evaluation

The proposed algorithm was implemented using Java running on a Windows PC with a 2.40 GHz Intel Core 2 Duo T8300 processor. To test our approach, we use CSPLib benchmark instances and real instances. To obtain the parameters (see Table 1) used in the experiments, the proposed BWAS algorithm was run 30 times on the benchmark and real instances and takes the values belonging to the best solutions reported.

*4.1. Benchmark Instances.* We analysed three benchmark instances where the performance of the algorithm was measured with good results with respect to speed of convergence, quality of the solution, and constraint satisfaction. For each instance of the problem, we performed 50 executions of the algorithm achieving the following results. First, we analysed the small instance of 8 periods (bacp8). This instance has 46

TABLE 1: Algorithm parameters.

Parameter	Value
Iterations	1000
Ants	8
Evaporation coefficient	0.2
$\alpha$	1
$\beta$	2
Mutation probability	0.3
Mutation power	0.5

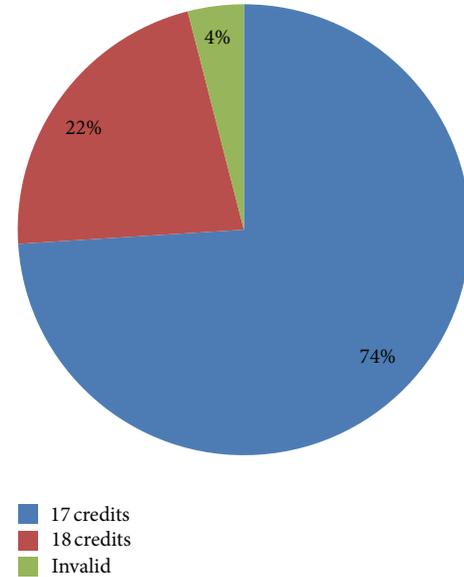


FIGURE 3: Number of credits for a small instance.

courses to be distributed in 8 academic periods. In Figure 3 we can see the results for bacp8 measured as a percentage considering 50 runs where 74% were able to find the optimal value. In a smaller percentage (4%) of cases, the case was an invalid solution. In most cases, our approach was able to find the optimal value.

The medium instance (bacp10) has 42 courses to be assigned to 10 periods, so the maximum load is smaller than bacp8. This means that the quality is better. After 50 runs the results are given in Figure 4. Again, the optimal solution was found in the majority of the cases.

The last instance is bacp12; in this case, where the complexity is greater, it has 66 courses to be assigned in 12 academic periods. In Figure 5, we can see that the algorithm found four different values and between 18 and 19 credits.

Of the 50 runs, we took the top 10 (not taking into account the invalid ones) and we summarize the results for the three instances shown in each column: name of the instance, best result, worst result, average, standard deviation, and optimal value known for the problem [6].

Also, to compare the times that the algorithm manages to find every quality of solution for each instance, we considered the best and worst solutions. Table 2 shows the difference between bacp12 and the other two instances, where the

TABLE 2: Result for benchmark instances.

Instance	BWAS Best	BWAS Worst	BWAS Average	$\sigma$	lp-solve Best	lp-solve Worst	Optimal
Bacp8	17	18	17.3	0.48	17	54	17
Bacp10	14	15	14.2	0.42	24	48	14
Bacp12	18	20	18.4	0.69	$\infty$	$\infty$	18

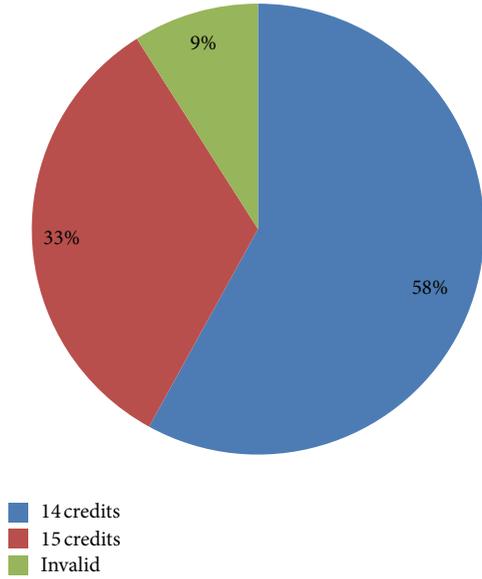


FIGURE 4: Number of credits for a medium instance.

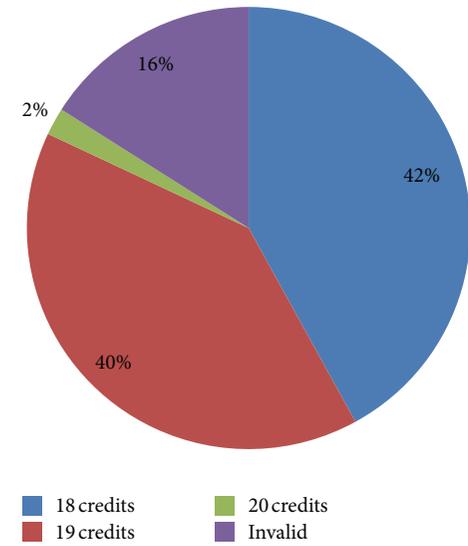


FIGURE 5: Number of credits for a large instance.

convergence time to the optimal solution is smaller in bacp8 and bacp10. But we cannot guarantee that 18 credits is the optimal solution in bacp12, but it is the best solution found by the algorithm. For the 12-period problem, lp-solve [3] does not get any log after a “turn-around-time” of 1 day.

In Figure 6, we show a graphic with the quality of the best solution found for each instance by iteration. The efficiency of

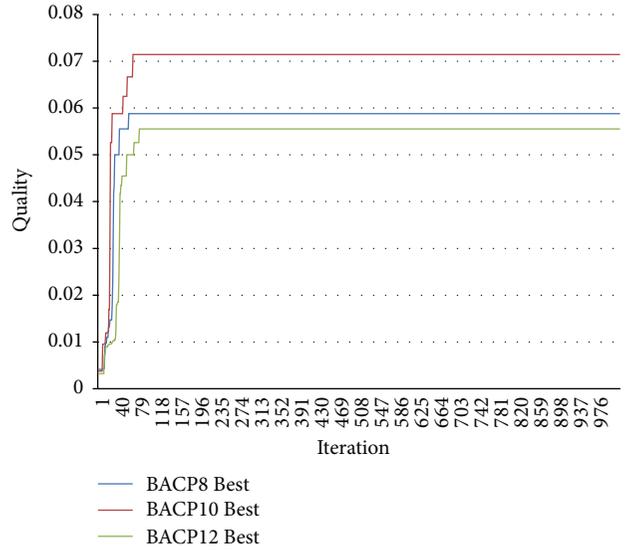


FIGURE 6: Efficiency of the algorithm (best solution).

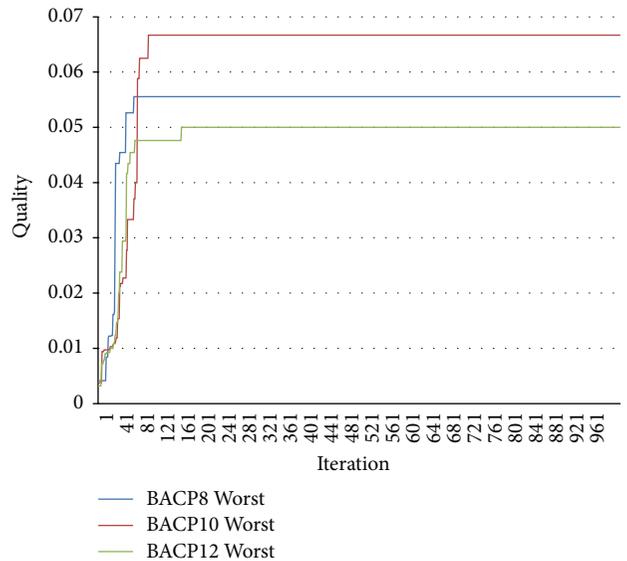


FIGURE 7: Efficiency of the algorithm (worst solution).

the algorithm is very high, as it converges before the first 100 iterations. In this graphic the quality is measured using (3). In the case of the worst solutions, no instances achieve the optimum, but they all converge quickly. The exception was bacp12 which takes time and moves away from the best value found, even more than the other two instances (see Figure 7).

TABLE 3: Time (ms) for the best solution to obtain the optimum.

Credits	Bacp8	Bacp10	Bacp12	Q (sol)
20	811	—	3994	0.05
19	—	—	5438	0.052
18	1045	1061	6368	0.055
17	1248	1342		0.058
16		1404		0.062
15		1451		0.066
14		1919		0.071

TABLE 4: Time (ms) for the worst solution to obtain the optimum.

Credits	Bacp8	Bacp10	Bacp12	Q (sol)
20	1201	1502	23448	0.05
19	1279	1532		0.052
18	1840	—		0.055
17		—		0.058
16		1602		0.062
15		2222		0.066

In Tables 3 and 4 we can see in the first column the maximum amount of credits of the curricula designed by the BWAS algorithm. The next three columns are the instances and the last column is the quality of the solution calculated using (3). The hyphen means that the quality of the solution was skipped. For example, in Table 4, the bacp10 instance went from a quality of 19 to a quality of 16. The empty spaces mean that the solution did not reach that value. The performance of lp-solve is very limited in contrast to the BWAS algorithm. In the case of the 8-period problem, the optimum plan is obtained in 1460 seconds. For the 10-period problem, no result is obtained after 5 hours, and the last result was logged before 1700 seconds (see additional details in [3]).

In Figure 7, we show the worst solution found by the BWAS algorithm for solving the BACP measured by quality.

**4.2. Real Instances.** To test our proposal with real instances, three benchmark files were created using the same format used in the test instances. The curriculums evaluated are Informatics Engineering, short program (8 periods), Informatics Engineering, long program (12 periods), from Catholic University of Valparaiso, and Informatics Engineering, medium program (10 periods), from Playa Ancha University. The academic plan of Informatics Engineering (short program, INF) has 34 courses; the complexity is lower than that of bacp8 because of the number of courses. Table 5 summarizes the results obtained (the optimal value is the maximum of the real curriculum for the career); for example, from 50 executions, the algorithm found a better solution than the real curricula (20 credits) with a maximum amount of 18 credits (see Figure 8).

Also, we evaluate the performance of the BWAS in the curriculum of Informatics Engineering (long program); in this instance the complexity is bigger than that of INF but lower than that of bacp12. It has 53 courses to be assigned to 12 academic periods. The results are shown in Figure 9.

TABLE 5: Result for real instances.

Instance	Best	Worst	Average	$\sigma$	Optimal
INF	18	18	18	0	20
ICI	18	18	18	0	18
UPLA	13	15	14	0.66	14

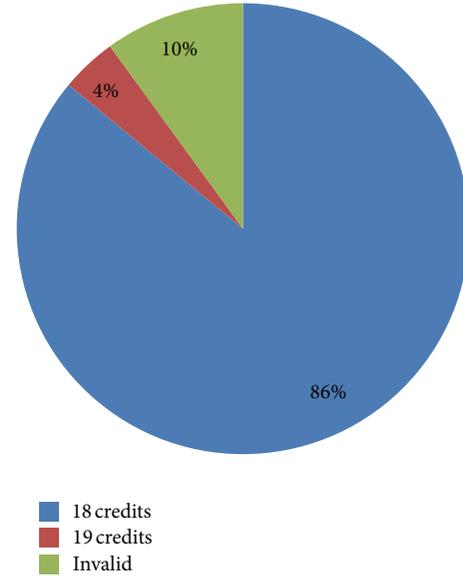


FIGURE 8: Number of credits for short real instance.

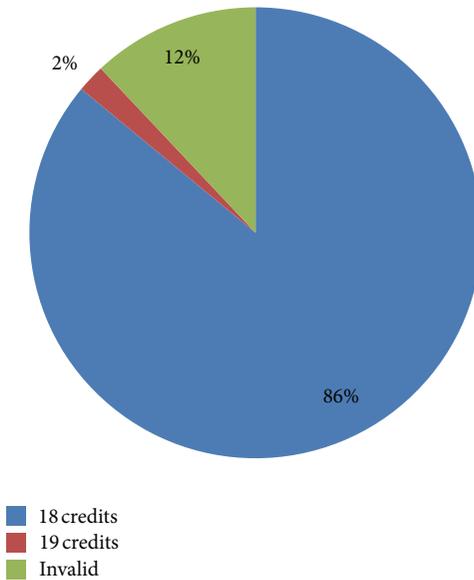


FIGURE 9: Number of credits for long real instance.

In this instance, the effectiveness was maintained with 86% of the runs, but the number of invalid solutions increases to 12%. The last evaluation corresponds to the Playa Ancha University instance (UPLA) where the optimal solution was found in only 20% of the iterations, surpassing the 14 credits of the official curriculum. However, the objective value was 14

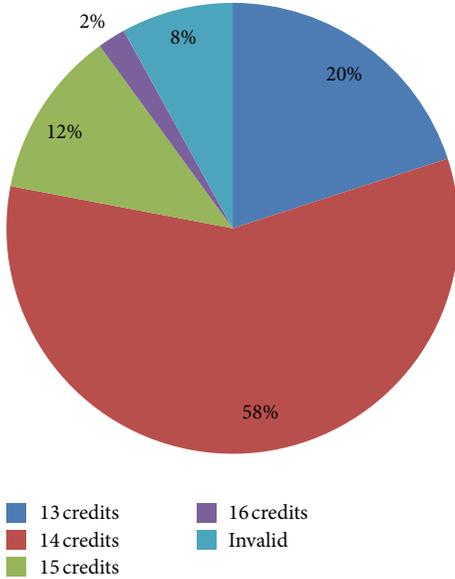


FIGURE 10: Number of credits for medium real instance.

TABLE 6: Time (ms) for the best solution to obtain the optimum in real instances.

Credits	INF	ICI	UPLA	Q (sol)
20	594	3484	—	0.05
19	—	3692	—	0.052
18	976	4080	—	0.055
17	—	—	—	0.058
16	—	—	2993	0.062
15	—	—	3078	0.066
14	—	—	3392	0.071

credits with 58% of the runs. This curriculum has 49 courses assigned in 10 semesters (see Figure 10). Again, we took the top 10 (regardless of the invalid solutions) from the 50 runs and summarized the results for the three instances.

Now we are going to compare the time and iteration in which the algorithm finds the best and worst solutions. In Table 6 we can see the time in milliseconds that the algorithm takes to reach the different solutions. In the case of INF and ICI, the time is very low. In the case of UPLA, it takes a little more than 10 seconds to reach the optimal value.

Figure 11 shows the quality of solution by iteration where we can see the speed of convergence.

In the case of the worst solutions (see Table 7), INF and ICI reach the optimum, but this time takes more seconds. In the case of UPLA, this instance did not reach the optimum but it takes less seconds.

In Figure 12 they see the worst solutions take a longer time to converge; that this is because the optimum value for the ICI and INF instances has the majority of the percentage. For UPLA the optimum is 13 credits, but the most repeated solution corresponds to 14 credits, which explains why the worst solution takes a longer time to converge. However, the

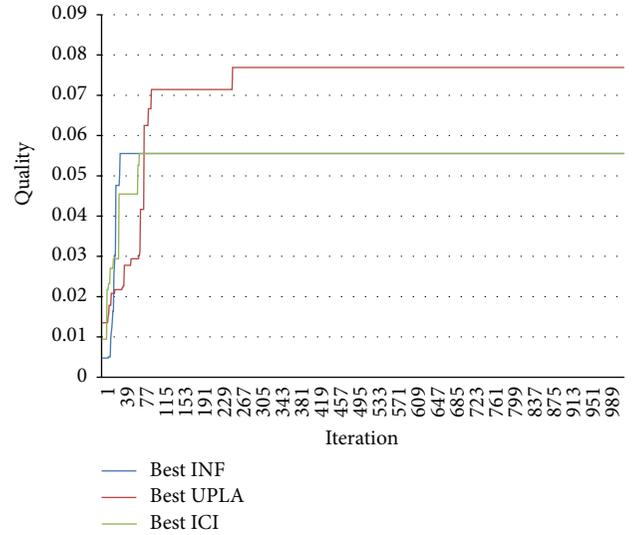


FIGURE 11: Efficiency of the algorithm in real instances (best solution).

TABLE 7: Time (ms) for the worst solution to obtain the optimum in real instances.

Credits	INF	ICI	UPLA	Q (sol)
20	3143	3969	—	0.05
19	3527	16671	—	0.052
18	8928	17274	—	0.055
17	—	—	3574	0.058
16	—	—	3682	0.062
15	—	—	3864	0.066
20	3143	3969	—	0.05

worst solution found by the algorithm has 15 credits for this instance.

## 5. Conclusions

In this work, we proposed an algorithm to solve the balanced academic curriculum problem using the ACO metaheuristic. The experimental evaluation shows the effectiveness of artificial ants solving constraint satisfaction problems. All the tests show that the margin of error is much smaller than the valid solutions delivered. For benchmark instances the quality of the solution was satisfactory and the optimal values of each instance were reached in the majority of the cases. In the real instances the results were also positive. In 2 instances (INF and UPLA) the results were better than in the official curricula and in the ICI results they were equal than in the official curricula. Moreover, the number of iterations to reach the optimum was low. Therefore, the BWAS algorithm is able to satisfy all constraints of the problem in most cases. Thus, we state that our proposal is a fairly reliable alternative to solve the BACP. For future work, we plan to tackle more complicated versions of BACP as was proposed in [1].

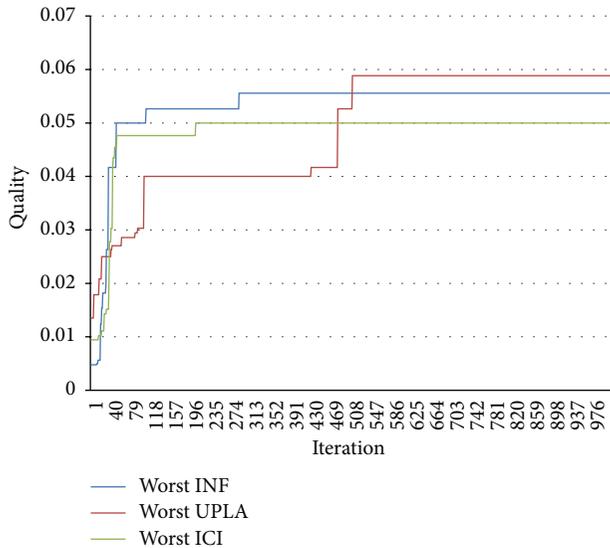


FIGURE 12: Efficiency of the algorithm in real instances (worst solution).

## Acknowledgments

Nibaldo Rodriguez is supported by Grant CONICYT/FONDECYT/Regular 1131105 and Ricardo Soto is supported by Grant CONICYT/FONDECYT/Iniciación 11130459.

## References

- [1] M. Chiarandini, L. D. Gaspero, S. Gualandi, and A. Schaerf, "The balanced academic curriculum problem revisited," *Journal of Heuristics*, vol. 18, no. 1, pp. 119–148, 2012.
- [2] J. N. Monette, P. Schaus, S. Zampelli, Y. Deville, and P. Dupont, "A cp approach to the balanced academic curriculum problem," in *Proceedings of the 7th International Workshop on Symmetry and Constraint Satisfaction Problems*, 2007.
- [3] C. Castro and S. Manzano, "Variable and value ordering when solving balanced academic curriculum problems," in *Proceedings of the 6th Workshop of the ERCIM WG on Constraints*, June 2001.
- [4] B. Hnich, Z. Kiziltan, and T. Walsh, "Modelling a balanced academic curriculum problem," in *Proceedings of the CP-AI-OR*, 2002.
- [5] C. Castro, B. Crawford, and E. Monfroy, "A quantitative approach for the design of academic curricula," in *Human Interface and the Management of Information. Interacting in Information Environments*, vol. 4558 of *Lecture Notes in Computer Science*, pp. 279–288, 2007.
- [6] T. Lambert, C. Castro, E. Monfroy, M. C. Riff, and F. Saubion, "Hybridization of genetic algorithms and constraint propagation for the BACP," in *Proceedings of the 21st International Conference on Logic Programming (ICLP '05)*, pp. 421–423, October 2005.
- [7] T. Lambert, C. Castro, E. Monfroy, and F. Saubion, "Solving the balanced academic curriculum problem with an hybridization of genetic algorithm and constraint propagation," in *Artificial Intelligence and Soft Computing—ICAISC 2006*, vol. 4029 of *Lecture Notes in Computer Science*, pp. 410–419, 2006.

- [8] M. Dorigo and C. Blum, "Ant colony optimization theory: a survey," *Theoretical Computer Science*, vol. 344, no. 2-3, pp. 243–278, 2005.
- [9] O. Cordon, I. F. de Viana, and F. Herrera, "Analysis of the best-worst ant system and its variants on the qap," in *Ant Algorithms*, pp. 228–234, 2002.
- [10] O. Cordon, I. F. de Viana, F. Herrera, and L. Moreno, "A new aco model integrating evolutionary computation concepts: the best-worst ant system," in *Ant Algorithms*, 2000.

## Research Article

# A Genetic-Algorithms-Based Approach for Programming Linear and Quadratic Optimization Problems with Uncertainty

Weihoa Jin, Zhiying Hu, and Christine W. Chan

*Energy Informatics Laboratory, Faculty of Engineering and Applied Science, University of Regina, Regina, SK, Canada S4S 0A2*

Correspondence should be addressed to Christine W. Chan; [christine.chan@uregina.ca](mailto:christine.chan@uregina.ca)

Received 18 June 2013; Accepted 22 October 2013

Academic Editor: Yudong Zhang

Copyright © 2013 Weihoa Jin et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper proposes a genetic-algorithms-based approach as an all-purpose problem-solving method for operation programming problems under uncertainty. The proposed method was applied for management of a municipal solid waste treatment system. Compared to the traditional interactive binary analysis, this approach has fewer limitations and is able to reduce the complexity in solving the inexact linear programming problems and inexact quadratic programming problems. The implementation of this approach was performed using the Genetic Algorithm Solver of MATLAB (trademark of MathWorks). The paper explains the genetic-algorithms-based method and presents details on the computation procedures for each type of inexact operation programming problems. A comparison of the results generated by the proposed method based on genetic algorithms with those produced by the traditional interactive binary analysis method is also presented.

## 1. Introduction

Economic optimization in operation planning of municipal solid waste management was first proposed in the 1960s [1]. Since then, different models of waste management planning have been proposed, which include linear programming [2–5], mixed integer linear programming, dynamic programming, multiobjective programming [6–8], and hybrids of these methods combined with probability, fuzzy, and inexact analyses [9–12]. The objectives of these waste management models include reduction of total cost, protection of the environment, and reuse of waste material and energy.

In real-life engineering problems, the available information often cannot be represented as deterministic numbers or distribution functions. Instead, it is often possible to represent the available information as inexact numbers, which can be readily used in the inexact programming models. This may be due to the fact that decision makers often prefer to have an inexact representation of uncertainty than provide a specification for distributions of fuzzy sets [13–17]. For operational planning, the inexact analysis approach typically treats the uncertain parameters as intervals with known upper and lower bounds but unclear distributions. A major

advantage of inexact analysis in operation planning is that variation of system performance and decision variables can be investigated by solving relatively simple submodels.

Research work on different kinds of inexact programming, such as inexact linear programming (ILP), inexact quadratic programming (IQP), inexact integer programming (IIP), inexact dynamic programming (IDP), and inexact multiobjective programming (IMOP) [6, 14, 15, 18–26], has been conducted. This paper presents an alternative heuristic-based method, which involves generic linear and quadratic programming with inexact information; the approach adopted involves the use of genetic algorithms (GA).

This paper is organized as follows. Section 2 presents the background of this research, which includes an introduction of the Genetic Algorithm Solver of MATLAB used for implementing the proposed method and the concepts of ILP, IQP, and their interactive binary analysis solution method [18]. Section 3 discusses the methodology of the proposed genetic-algorithms-based methods for solving inexact linear problems and inexact quadratic problems. Section 4 presents the solution of the IQP problem of solid waste disposal planning as a case study.

## 2. Background

Linear programming and nonlinear programming are considered powerful optimization tools suitable for modeling and solving complex optimization problems in engineering. To handle uncertainty in real world data, inexact parameters and constraints are combined with various kinds of optimization techniques. Huang et al. [18–21] proposed two inexact nonlinear programming methods by introducing internal and fuzzy numbers into the quadratic programming (QP) frameworks. The methods of inexact quadratic programming (IQP) and inexact-fuzzy quadratic programming are applicable for operation planning of solid waste management systems. Often a detailed solution of IQP involves a large number of direct comparisons to interactively identify the uncertain relationships among the objective function and decision variables, whether the problems are medium-sized or larger-scaled. When these methods are applied to complicated and nonlinear problems, the number of direct comparisons can become exponential. In such a situation, we suggest that GA are a feasible problem-solving method.

GA have been applied as the optimization techniques for solving complex and nonlinear problems in operations research, industrial engineering, and management science. The GA method is a suitable optimization tool especially for solving problems, which involve nonsmooth and multimodal search spaces. An engineering problem that has traditionally been solved as an IQP problem often involves a large and uneven search space, for which a global optimal solution is often not required. Instead, we suggest that the GA-based method is a more effective problem-solving approach than some traditional inexact programming methods.

*2.1. Genetic Algorithm Solver for MATLAB.* For implementation of genetic algorithms, the Genetic Algorithm Solver of Global Optimization Toolbox (GASGOT), developed by MATLAB (Trademark of MathWorks), has been adopted. GASGOT implements simulated evolution in the MATLAB environment using both binary and floating point representations and ordered base representation. This enables flexible implementation of the genetic operators, selection functions, termination functions, and evaluation functions. GASGOT was developed by the Department of Industrial Engineering of North Carolina State University as a toolbox of MATLAB. Hence, it runs in a MATLAB workspace and can be easily invoked by other programs. GASGOT supports implementation of binary chromosomes, binary mutation, and simple crossover. For floating point representation, the operators of uniform mutation, nonuniform mutation, multinonuniform mutation, boundary mutation, simple crossover, arithmetic crossover, and heuristic crossover are defined.

The GASGOT is adopted as the problem-solving engine of both the GA linear program and GA nonlinear program; all the applications and numeric examples were calculated using this solver in MATLAB.

*2.2. Inexact Linear Programming and Its Problem-Solving Approach.* To support decision making involving uncertainties, Huang et al. [20, 22] proposed an interactive binary

analysis to solve the inexact linear programming (ILP) problem.

A typical ILP problem can be expressed as follows:

$$\begin{aligned} \text{Max } f^\pm &= \sum_{j=1}^n [c_j^\pm x_j^\pm], \\ \text{s.t. } \sum_{j=1}^n a_{ij}^\pm x_j^\pm &\leq b_i^\pm, \quad i = 1, 2, \dots, m, \\ x_j^\pm &\geq 0, \quad j = 1, 2, \dots, n, \end{aligned} \quad (1)$$

where  $a_{ij}^\pm$ ,  $b_i^\pm$ , and  $c_j^\pm$  are inexact parameters and  $x_j^\pm$  is an inexact variable. It is assumed that an optimal solution exists. For an inexact number  $g^\pm \in [g^-, g^+]$ ,  $g^+$  and  $g^-$  are the upper and lower bounds, respectively.

The traditional binary solution procedure is specified as follows.

*Step 1.* Group symbols for inexact coefficients  $c_j^\pm$ ; let former  $k_1$  coefficients be positive and latter  $k_2$  be negative;  $k_1 + k_2 = n$ .

*Step 2.* Define the upper and lower bounds of the objective function as  $f^+$  and  $f^-$ .

*Step 3.* Define absolute values and signs for the coefficients of the constraints  $a_{ij}^\pm$ .

*Step 4.* Define the relationships between the decision variables  $x_j^\pm$  and the absolute value of the coefficients of the constraints  $|a_{ij}^\pm|$ .

*Step 5.* Formulate constraints corresponding to the upper and lower bounds of the objective function  $f^+$  and  $f^-$ .

*Step 6.* When the right-hand side of the constraints  $b_i$  are also inexact numbers, define the relationships between  $f^\pm$  and  $b_i^\pm$ .

*Step 7.* Specify the two submodels.

For a detailed description of the procedure, see Huang et al. [20, 22, 23].

*2.3. Inexact Quadratic Programming and Its Problem-Solving Approach.* A typical IQP problem is formulated as follows:

$$\begin{aligned} \text{Max } f^\pm &= \sum_{j=1}^n [c_j^\pm x_j^\pm + d_j^\pm (x_j^\pm)^2], \\ \text{s.t. } \sum_{j=1}^n a_{ij}^\pm x_j^\pm &\leq b_i^\pm, \quad i = 1, 2, \dots, m, \\ x_j^\pm &\geq 0, \quad j = 1, 2, \dots, n, \end{aligned} \quad (2)$$

where  $a_{ij}^\pm$ ,  $b_i^\pm$ ,  $c_j^\pm$ , and  $d_j^\pm$  are inexact parameters,  $x_j^\pm$  is an inexact variable, and it is assumed that an optimal solution exists.

The solution procedure is similar to that of ILP but involves more complexity and computation; the main steps of the solution procedure are listed as follows; for a detailed description, see Huang et al. [18, 24, 25].

*Step 1.* Group symbols for inexact coefficients  $c_j^\pm$  and  $d_{ij}^\pm$ ; when  $c_j^\pm$  and  $d_{ij}^\pm$  have the same signs, similar to the ILP, let former  $k_1$  coefficients be positive and latter  $k_2$  be negative;  $k_1 + k_2 = n$ . When  $c_j^\pm$  and  $d_{ij}^\pm$  have different signs,  $2^n$  combinations of the upper and lower bounds of  $x_j^\pm$  have to be formulated for the objective function, which will require a large number of computations.

*Step 2.* Define the upper and lower bounds of the objective function as  $f^+$  and  $f^-$ .

*Step 3.* Define the absolute values and signs for the coefficients of the constraints  $a_{ij}^\pm$  and the relationships between the decision variables  $x_j^\pm$  and the absolute value of the coefficients of the constraints  $|a_{ij}^\pm|$ .

When some  $x_j^-$  corresponds to  $f^+$  and some  $x_j^+$  corresponds to  $f^-$ , the specification of the constraints  $a_{ij}^\pm x_j^\pm \leq b_i^\pm$  requires a comparison of the contribution of  $x_j^+$  and  $x_j^-$  groups to the sum  $\sum_{j=1}^n a_{ij}^\pm x_j^\pm$ , when  $f^+$  is desired. When the problem is complex, a direct comparison of the dominance of  $x_j^+$  and  $x_j^-$  becomes impossible; then some simplification and assumption need to be considered, which will affect the quality of the result.

*Step 4.* Formulate the constraints corresponding to the upper and lower bounds of the objective function  $f^+$  and  $f^-$ .

*Step 5.* When the right-hand side of the constraints  $b_i$  are also inexact numbers, define the relationships between  $f^\pm$  and  $b_i^\pm$ .

*Step 6.* Specify the two submodels.

The genetic-algorithms-based methods to solve the above inexact linear problem and inexact quadratic problem will be presented in the next section, and the results from the GA-based methods will be compared to those generated using the traditional approach in [18, 20, 22–25].

### 3. Methodology

*3.1. Genetic-Algorithms-Based Method for Solution of ILP Problems (GAILP).* A GA, as a heuristic search algorithm, has been adopted for solving the aforementioned ILP problem. In the GA approach, the upper and lower bounds of the inexact numbers of coefficients  $a_{ij}^\pm$ ,  $b_i^\pm$ , and  $c_j^\pm$  can be determined by substituting the initial suboptimal decision variables into the objective function.  $f^+$  and  $f^-$  can be calculated directly without any uncertainty in the coefficients. This approach is called the genetic-algorithms-based method for solving ILP problems or the GAILP method.

GAILP has been designed to include three stages, which are discussed as follows.

The objective of the first stage is to get an initial suboptimal  $x_j^s$  for the following problem, which is transformed from the ILP problem defined in (1):

$$\begin{aligned} \text{Max } f^\pm &= \sum_{j=1}^n [c_j^r x_j^s], \\ \text{s.t. } \sum_{j=1}^n a_{ij}^r x_j^s &\leq b_i^r, \quad i = 1, 2, \dots, m, \\ x_j &\geq 0, \quad j = 1, 2, \dots, n, \end{aligned} \quad (3)$$

where  $a_{ij}^r$ ,  $b_i^r$ , and  $c_j^r$  are random numbers that satisfy the continuous uniform distribution in the intervals of  $[a_{ij}^-, a_{ij}^+]$ ,  $[b_i^-, b_i^+]$ , and  $[c_j^-, c_j^+]$ , respectively. Then, the problem is solved by the GA linear program solving engine of GASGOT, which uses the objective function in (3) as the positive term of the fitness function and the constraints of (1) as the negative punishment terms. Thus, a suboptimal solution  $f^s$  can be identified and the corresponding decision variables of  $x_j^s$  are also obtained.

In the second stage, the inexact coefficients of  $a_{ij}^\pm$ ,  $b_i^\pm$ , and  $c_j^\pm$  will be determined. Let the determined coefficients corresponding to  $f^+$  be  $a_{ij}^{++}$ ,  $b_i^{++}$ , and  $c_j^{++}$  and those corresponding to  $f^-$  be  $a_{ij}^{--}$ ,  $b_i^{--}$ , and  $c_j^{--}$ . These two sets of coefficients can be obtained using the following method.

Substituting  $x_j^s$  into (1) will convert (1) into the following equation:

$$\begin{aligned} \text{Max } f^\pm &= \sum_{j=1}^n [c_j^\pm x_j^s], \\ \text{s.t. } \sum_{j=1}^n a_{ij}^\pm x_j^s &\leq b_i^\pm, \quad i = 1, 2, \dots, m, \\ x_j^s &\geq 0, \quad j = 1, 2, \dots, n. \end{aligned} \quad (4)$$

To identify the coefficients  $a_{ij}^\pm$ ,  $b_i^\pm$ , and  $c_j^\pm$  corresponding to  $f^\pm$ , a set of objective functions needs to be constructed and solved. Since  $x_j^s$  are suboptimal variables, which tend to make the objective function closer to  $f^+$ , consider  $a_{ij}^\pm$ ,  $b_i^\pm$ , and  $c_j^\pm$  as variables; then the objective function of (5) can be constructed so as to find  $c_j^{\pm+}$ :

$$\begin{aligned} \text{Max } f^\pm &= \sum_{j=1}^n [c_j^\pm x_j^s], \\ \text{s.t. } \sum_{j=1}^n a_{ij}^\pm x_j^s &\leq b_i^\pm, \quad i = 1, 2, \dots, m. \end{aligned} \quad (5)$$

The coefficients  $c_j^{\pm+}$  are considered as corresponding to  $f^+$ .

At the same time, the objective function presented in (6) can be constructed so as to find  $c_j^{\pm}$ :

$$\begin{aligned} \text{Min } f^{\pm} &= \sum_{j=1}^n [c_j^{\pm} x_j^s], \\ \text{s.t. } \sum_{j=1}^n a_{ij}^{\pm} x_j^s &\leq b_i^{\pm}, \quad i = 1, 2, \dots, m. \end{aligned} \quad (6)$$

There are two kinds of decision schemes for inexact programming problems, which are the conservative schemes and optimistic schemes [26]. The former assumes less risk than the latter, so that, for a maximization objective function, planning for the lower bound of an objective value represents the conservative scheme, and planning for the upper bound of an objective value represents the optimistic scheme [26]. In terms of constraints, the conservative scheme involves more rigorous or stringent constraints, and the optimistic scheme adopts more tolerant ones.

Thus, the problem of searching for  $a_{ij}^{\pm+}$  and  $b_i^{\pm+}$  of the optimistic scheme and corresponding to the upper bound of the objective value of  $f^+$  can be represented as follows:

$$\begin{aligned} \text{Max } \sum_{j=1}^n \text{abs}(a_{ij}^{\pm} x_j^s - b_i^{\pm}), \\ \text{s.t. } \sum_{j=1}^n a_{ij}^{\pm} x_j^s &\leq b_i^{\pm}, \quad i = 1, 2, \dots, m. \end{aligned} \quad (7)$$

The problem

$$\begin{aligned} \text{Min } \sum_{j=1}^n \text{abs}(a_{ij}^{\pm} x_j^s - b_i^{\pm}), \\ \text{s.t. } \sum_{j=1}^n a_{ij}^{\pm} x_j^s &\leq b_i^{\pm}, \quad i = 1, 2, \dots, m, \end{aligned} \quad (8)$$

will give  $a_{ij}^{\pm-}$  and  $b_i^{\pm-}$  of the conservative scheme, corresponding to the lower bound of the objective value of  $f^-$ .

Hence, the values of  $a_{ij}^{\pm+}$ ,  $b_i^{\pm+}$ , and  $c_j^{\pm+}$  and  $a_{ij}^{\pm-}$ ,  $b_i^{\pm-}$ , and  $c_j^{\pm-}$  can be calculated.

In the third stage, the problem represented in (1) is converted into the following two subproblems.

For  $f^+$ ,

$$\begin{aligned} \text{Max } f^+ &= \sum_{j=1}^n [c_j^{\pm+} x_j^{\pm}], \\ \text{s.t. } \sum_{j=1}^n a_{ij}^{\pm+} x_j^{\pm} &\leq b_i^{\pm+}, \quad i = 1, 2, \dots, m, \\ x_j^{\pm} &\geq 0, \quad j = 1, 2, \dots, n. \end{aligned} \quad (9)$$

For  $f^-$ ,

$$\begin{aligned} \text{Max } f^- &= \sum_{j=1}^n [c_j^{\pm-} x_j^{\pm}], \\ \text{s.t. } \sum_{j=1}^n a_{ij}^{\pm-} x_j^{\pm} &\leq b_i^{\pm-}, \quad i = 1, 2, \dots, m, \\ x_j^{\pm} &\geq 0, \quad j = 1, 2, \dots, n. \end{aligned} \quad (10)$$

This step eliminates the inexact parameters in (1) and generates instead (9) and (10) as typical linear programming (LP) problems, which can be solved easily using the traditional methods. Generally speaking, the interactive binary algorithm of Huang et al. [20, 22] can be used for solving inexact linear problems reliably and relatively quickly for many real-life decision-making scenarios in the engineering field. However, this binary algorithm has some limitations. One of them, for example, is the limitation that the upper and lower bounds of an inexact coefficient cannot have different signs. By contrast, the GAILP does not have this kind of limitation because the GA method does not depend on any assumed distribution of the inexact parameter.

The following inexact linear problem demonstrates how GAILP is able to handle this situation:

$$\begin{aligned} \text{Max } f^{\pm} &= [2, 6] x_1^{\pm} + [7.8, 10.2] x_2^{\pm}, \\ \text{s.t. } x_1^{\pm} + [-0.8, 1.2] x_2^{\pm} &\leq 4.6, \\ [-6, -8.5] x_1^{\pm} + 12x_2^{\pm} &\leq [-2, 3], \\ x_1^{\pm} &\geq 0, \quad x_2^{\pm} \geq 0. \end{aligned} \quad (11)$$

In stage one, the suboptimal  $f^s$  and the corresponding  $x_1^s$  and  $x_2^s$  are calculated:

$$f^s = 112, \quad x_1^s = 10.1, \quad x_2^s = 5.8. \quad (12)$$

In stage two, substituting  $x_1^s$  and  $x_2^s$  into (11), the two sets of problems for determining  $a_{ij}^{\pm+}$ ,  $b_i^{\pm+}$ , and  $c_j^{\pm+}$  and  $a_{ij}^{\pm-}$ ,  $b_i^{\pm-}$ , and  $c_j^{\pm-}$  are constructed for  $f^+$  and  $f^-$ , respectively.

For  $f^+$ ,  $c_j^{\pm+}$  can be determined by solving the following problem:

$$\begin{aligned} \text{Max } f^+ &= c_1^{\pm+} 10.1 + c_2^{\pm+} 5.8, \\ \text{s.t. } 10.1 + [-0.8, 1.2] 5.8 &\leq 4.6, \\ [-6, -8.5] 10.1 + 125.8 &\leq [-2, 3], \\ c_1^{\pm+} &\leq [2, 6], \quad c_2^{\pm+} \leq [7.8, 10.2]. \end{aligned} \quad (13)$$

And  $a_{ij}^{\pm+}$  and  $b_i^{\pm+}$  can be determined by solving the following problem:

$$\begin{aligned} & \text{Max } 4.6 - (x_1^s + a_{12}x_2^s), \\ & \text{Max } b_2 - (a_{21}x_1^s + 12x_2^s), \\ & a_{12} = [-0.8, 1.2], \\ & a_{21} = [-6, -8.5], \\ & b_2 = [-2, 3]. \end{aligned} \quad (14)$$

For  $f^-, c_j^{\pm-}$  can be determined by solving the following problem:

$$\begin{aligned} & \text{Min } f^- = c_1^{\pm-} 10.1 + c_2^{\pm-} 5.8, \\ & \text{s.t. } 10.1 + [-0.8, 1.2] 5.8 \leq 4.6, \\ & [-6, -8.5] 10.1 + 125.8 \leq [-2, 3], \\ & c_1^{\pm-} \leq [2, 6], \quad c_2^{\pm-} \leq [7.8, 10.2]. \end{aligned} \quad (15)$$

And  $a_{ij}^{\pm-}$  and  $b_i^{\pm-}$  can be determined by solving the following problem:

$$\begin{aligned} & \text{Min } 4.6 - (x_1^s + a_{12}x_2^s), \\ & \text{Min } b_2 - (a_{21}x_1^s + 12x_2^s), \\ & a_{12} = [-0.8, 1.2], \\ & a_{21} = [-8.5, -6], \\ & b_2 = [-2, 3]. \end{aligned} \quad (16)$$

By solving (13), (14), (15), and (16), the coefficients are calculated as follows:

$$\begin{aligned} a_{12}^{\pm+} &= -0.8, & a_{21}^{\pm+} &= -8.5, & b_2^{\pm+} &= 3, \\ c_1^{\pm+} &= 6, & c_2^{\pm+} &= 10.2, \\ a_{12}^{\pm-} &= 1.2, & a_{21}^{\pm-} &= -6, & b_2^{\pm-} &= -2, \\ c_1^{\pm-} &= 2, & c_2^{\pm-} &= 7.8. \end{aligned} \quad (17)$$

In stage three, the two submodels for the optimistic scheme and the conservative scheme are constructed, respectively, as follows.

For the optimistic scheme,  $f^+$ ,

$$\begin{aligned} & \text{Max } f^+ = 6x_1^+ + 10.2x_2^+, \\ & \text{s.t. } x_1^+ - 0.8x_2^+ \leq 4.6, \\ & -8.5x_1^+ + 12x_2^+ \leq 3, \\ & x_1^+ \geq 0, \quad x_2^+ \geq 0. \end{aligned} \quad (18)$$

For the conservative scheme,  $f^-$ ,

$$\begin{aligned} & \text{Max } f^- = 2x_1^- + 7.8x_2^-, \\ & \text{s.t. } x_1^- + 1.2x_2^- \leq 4.6, \\ & -6x_1^- + 12x_2^- \leq 2, \\ & x_1^- \geq 0, \quad x_2^- \geq 0. \end{aligned} \quad (19)$$

The final result can be found by solving the previous two submodels:

$$\begin{aligned} f^+ &= 149.04, & x_1^+ &= 11.08, & x_2^+ &= 8.1, \\ f^- &= 16.4, & x_1^- &= 3, & x_2^- &= 1.33. \end{aligned} \quad (20)$$

From the previous calculation, it can be seen that the GAILP method can be used without any assumption of the upper and lower bounds of the inexact coefficients. In fact, this method effectively extends the scope of problems solvable using the methods of the inexact linear programming problem. Therefore, the GAILP method is more adaptable for real world applications of optimization problems with uncertainty. In the next section, this method will be extended to solve the inexact quadratic problems.

A sample inexact linear programming problem in [22] is as follows:

$$\begin{aligned} & \text{Max } f^\pm = c_1x_1^\pm + c_2x_2^\pm, \\ & \text{s.t. } a_{11}x_1^\pm + a_{12}x_2^\pm \leq b_1, \\ & a_{21}x_1^\pm + a_{22}x_2^\pm \leq b_2, \end{aligned} \quad (21)$$

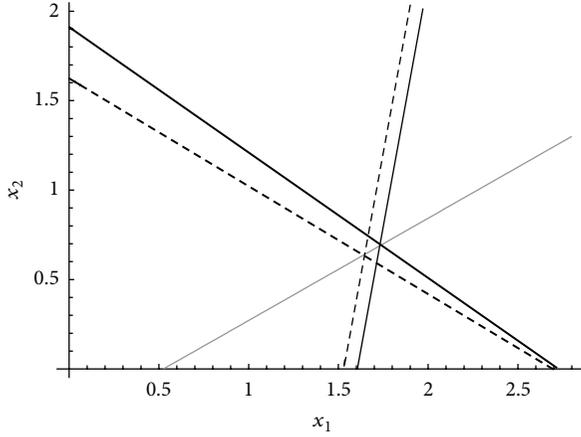
where  $c_1 = [26, 30]$ ,  $c_2 = [-6, -5.5]$ ,

$$\begin{aligned} a_{11} &= [8, 10], & a_{12} &= [-14, -12], & b_1 &= [3.8, 4.2], \\ a_{21} &= [2.4, 2.8], & a_{22} &= [3.4, 4], & b_2 &= 6.5. \end{aligned} \quad (22)$$

By using the traditional interactive binary algorithm, two submodels are obtained:

$$\begin{aligned} & \text{Max } f^+ = 30x_1^+ - 5.5x_2^-, \\ & \text{s.t. } 8x_1^+ - 14x_2^- \leq 4.2, \\ & 2.4x_1^+ + 4x_2^- \leq 6.5, \\ & x_1^+ \geq 0, \quad x_2^- \geq 0, \\ & \text{Max } f^- = 26x_1^- - 6.0x_2^+, \\ & \text{s.t. } 10x_1^- - 12x_2^+ \leq 3.8, \\ & 2.8x_1^- + 3.4x_2^+ \leq 6.5, \\ & x_1^- \geq 0, \quad x_2^+ \geq 0. \end{aligned} \quad (23)$$

The result was  $f^+ = 45.78$ ,  $x_1 = 1.64$ , and  $x_2 = 0.64$ ;  $f^- = 30.77$ ,  $x_1 = 1.37$ , and  $x_2 = 0.79$ .

FIGURE 1: Optimistic scheme,  $f^+$ .

For a detailed description of the problem, see [22].

By using the GAILP method as stated in (21), the result can be calculated with the following objective functions:

$$\begin{aligned}
 \text{Max } & f^+ = 30x_1^+ - 5.5x_2^+, \\
 \text{s.t. } & 8x_1^+ - 14x_2^+ \leq 4.2, \\
 & 2.4x_1^+ + 3.4x_2^+ \leq 6.5, \\
 & x_1^+ \geq 0, \quad x_2^+ \geq 0, \\
 \text{Max } & f^- = 26x_1^- - 6.0x_2^-, \\
 \text{s.t. } & 10x_1^- - 12x_2^- \leq 3.8, \\
 & 2.8x_1^- + 4x_2^- \leq 6.5, \\
 & x_1^- \geq 0, \quad x_2^- \geq 0.
 \end{aligned} \tag{24}$$

The result was  $f^+ = 48.15$ ,  $x_1 = 1.73$ , and  $x_2 = 0.69$ ;  $f^- = 29.15$ ,  $x_1 = 1.29$ , and  $x_2 = 0.72$ .

The GAILP method generates a solution, which is different from that obtained using the interactive binary analysis proposed in [22]. A comparison of the results will be discussed as follows.

For the  $f^+$  optimistic scheme, the GAILP method can generate a result that is guaranteed to be as close as possible to the upper bound of the constraints. Hence, the maximized value of the objective function is greater than that produced by the interactive binary analysis. For the  $f^-$  conservative scheme, the GAILP method has a higher probability of satisfying the requirements of the constraints as close as possible to the lowest limit. Hence, the maximized objective value is smaller.

In Figures 1, 2, 3, and 4 the bold lines denote the boundaries of the constraints, which limit the possible values for  $x_1$  and  $x_2$  to the left lower area. The constraint  $a_{11}x_1^+ + a_{12}x_2^+ \leq b_1$  is shown in these figures as the grey bold solid lines, which is the same for both the interactive binary analysis and the GAILP method. The dark bold dotted lines represent the constraint of  $a_{21}x_1^+ + a_{22}x_2^+ \leq b_2$  given by the interactive

TABLE 1: Legend for Figures 1 to 4.

-----	The constraint $a_{21}x_1^+ + a_{22}x_2^+ \leq b_2$ given by interactive binary analysis
————	The constraint $a_{21}x_1^+ + a_{22}x_2^+ \leq b_2$ given by GAILP
————	The constraint $a_{11}x_1^+ + a_{12}x_2^+ \leq b_1$
-----	Objective function line given by interactive binary analysis
————	Objective function line given by GAILP

binary analysis, and the dark bold solid lines represent the same constraint given by the proposed GAILP method.

The boundaries, together with the  $x_1$  and  $x_2$  axes, enclose the entire area defined by the constraints. The objective functions  $f^+ = 30x_1^+ - 5.5x_2^+$  or  $f^- = 26x_1^- - 6x_2^-$  are groups of parallel lines, as shown in the figures by the thin solid and dotted lines. According to different values of  $x_1$  and  $x_2$ , these objective function lines would have different intercepts on both axes. These constraints restrict the objective function lines to cross with the constraints area, so that, at some vertex, the objective function would reach its extreme (i.e., maximized or minimized) values.

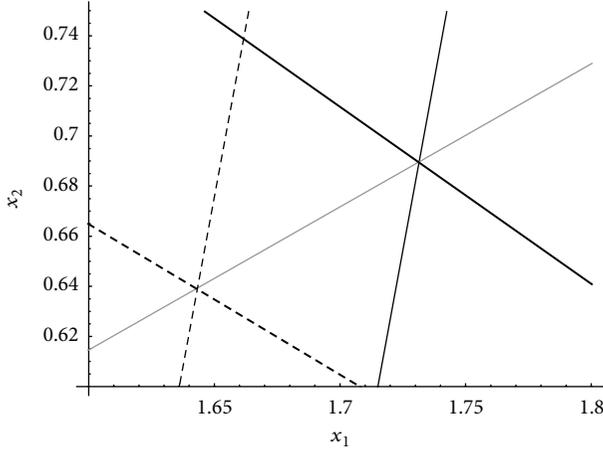
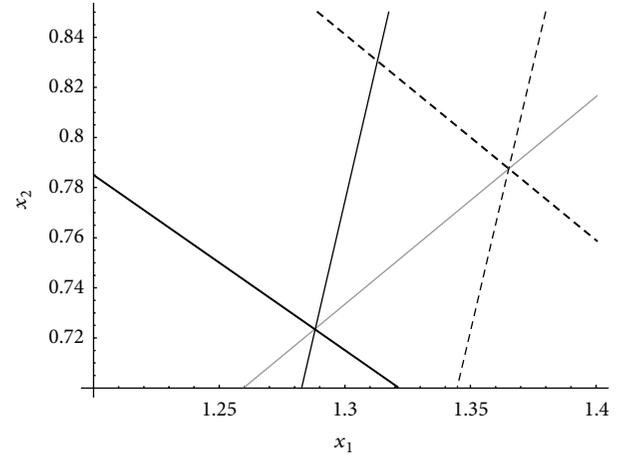
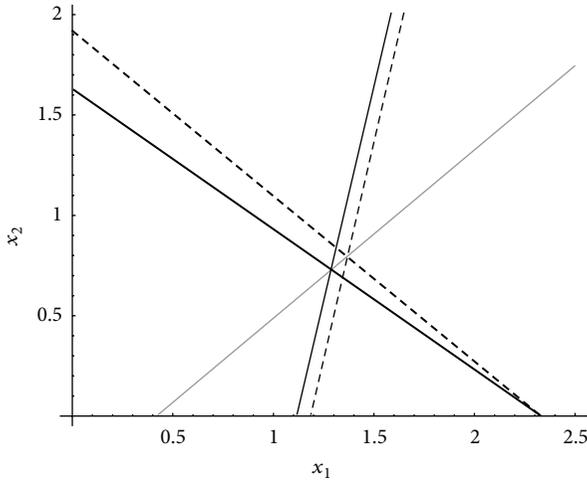
In Figures 1, 2, 3, and 4, the thin dotted lines are given by the interactive binary analysis, and the thin solid lines represent the objective functions given by the proposed GAILP method. The legend for Figure 1 to Figure 4 is shown in Table 1.

**3.2. Genetic-Algorithms-Based Method for Solving IQP Problems (GAIQP).** The GAILP method can be extended to solve the inexact quadratic programming (IQP) problems or other more complicated inexact nonlinear programming problems. The typical IQP problem was presented in Section 2 as (2).

In stage one, to obtain an initial suboptimal  $x_j^s$  from a problem transformed from the IQP problem we use the following:

$$\begin{aligned}
 \text{Max } & f = \sum_{j=1}^n \left[ c_j^r x_j + d_j^r (x_j)^2 \right], \\
 \text{s.t. } & \sum_{j=1}^n a_{ij}^r x_j^r \leq b_j^r, \quad i = 1, 2, \dots, m, \\
 & x_j \geq 0, \quad j = 1, 2, \dots, n,
 \end{aligned} \tag{25}$$

where  $a_{ij}^r$ ,  $b_i^r$ ,  $c_j^r$ , and  $d_j^r$  are random numbers that satisfy the continuous uniform distribution in the intervals  $[a_{ij}^-, a_{ij}^+]$ ,  $[b_i^-, b_i^+]$ ,  $[c_j^-, c_j^+]$ , and  $[d_j^-, d_j^+]$ . Then, a suboptimal solution  $f^s$  can be identified, and the corresponding decision variables  $x_j^s$  are also obtained.


 FIGURE 2: Zoom-in of the optimistic scheme,  $f^+$ .

 FIGURE 4: Zoom-in of the conservative scheme,  $f^-$ .

 FIGURE 3: Conservative scheme,  $f^-$ .

In the second stage, substituting  $x_j^s$  into the formula in (2) converts (2) into the following formula:

$$\begin{aligned} \text{Max } f^\pm &= \sum_{j=1}^n [c_j^\pm x_j^s + d_j^\pm (x_j^s)^2], \\ \text{s.t. } \sum_{j=1}^n a_{ij}^\pm x_j^s &\leq b_i^\pm, \quad i = 1, 2, \dots, m, \\ x_j^s &\geq 0, \quad j = 1, 2, \dots, n. \end{aligned} \quad (26)$$

To determine the coefficients  $a_{ij}^\pm$ ,  $b_i^\pm$ ,  $c_j^\pm$ , and  $d_j^\pm$  corresponding to  $f^\pm$  we use the following:

$$\begin{aligned} \text{Max } f^\pm &= \sum_{j=1}^n [c_j^\pm x_j^s + d_j^\pm (x_j^s)^2], \\ \text{s.t. } \sum_{j=1}^n a_{ij}^\pm x_j^s &\leq b_i^\pm, \quad i = 1, 2, \dots, m, \end{aligned} \quad (27)$$

$$\text{Min } f^\pm = \sum_{j=1}^n [c_j^\pm x_j^s + d_j^\pm (x_j^s)^2], \quad (28)$$

$$\text{s.t. } \sum_{j=1}^n a_{ij}^\pm x_j^s \leq b_i^\pm, \quad i = 1, 2, \dots, m.$$

To determine  $a_j^{\pm+}$  and  $b_i^{\pm+}$  of the optimistic scheme corresponding to the upper limit of the objective value of  $f^+$ , we have the following:

$$\text{Max } \sum_{j=1}^n \text{abs}(a_{ij}^\pm x_j^s - b_i^\pm), \quad (29)$$

$$\text{s.t. } \sum_{j=1}^n a_{ij}^\pm x_j^s \leq b_i^\pm, \quad i = 1, 2, \dots, m.$$

To obtain  $a_j^{\pm-}$  and  $b_i^{\pm-}$ , we use the following:

$$\text{Min } \sum_{j=1}^n \text{abs}(a_{ij}^\pm x_j^s - b_i^\pm), \quad (30)$$

$$\text{s.t. } \sum_{j=1}^n a_{ij}^\pm x_j^s \leq b_i^\pm, \quad i = 1, 2, \dots, m.$$

In the third stage, the problem expressed in (2) has been converted into the following two subproblems.

For  $f^+$ ,

$$\text{Max } f^+ = \sum_{j=1}^n [c_j^{\pm+} x_j^\pm + d_j^{\pm+} (x_j^\pm)^2],$$

$$\text{s.t. } \sum_{j=1}^n a_{ij}^{\pm+} x_j^\pm \leq b_i^{\pm+}, \quad i = 1, 2, \dots, m, \quad (31)$$

$$x_j^\pm \geq 0, \quad j = 1, 2, \dots, n.$$

For  $f^-$ ,

$$\begin{aligned} \text{Max } f^- &= \sum_{j=1}^n [c_j^{\pm-} x_j^{\pm} + d_j^{\pm-} (x_j^{\pm})^2], \\ \text{s.t. } \sum_{j=1}^n a_{ij}^{\pm-} x_j^{\pm} &\leq b_i^{\pm-}, \quad i = 1, 2, \dots, m, \\ x_j^{\pm} &\geq 0, \quad j = 1, 2, \dots, n. \end{aligned} \quad (32)$$

The inexact information has been incorporated in these two subproblems. These two subproblems, as typical nonlinear programming problems, can be solved by the GA nonlinear program solver engine of GASGOT.

This method is applied to an IQP problem that was originally proposed by [27]. This IQP problem can be expressed as follows:

$$\begin{aligned} \text{Max } f^{\pm} &= [16, 18] x_1^{\pm} - [12, 14] (x_1^{\pm})^2 \\ &\quad - [4, 5] x_2^{\pm} + [14, 15] (x_2^{\pm})^2, \\ \text{s.t. } [4.5, 5.5] x_1^{\pm} + [1.8, 2.2] x_2^{\pm} &\leq [1.8, 2.1], \\ x_1^{\pm} + [1.8, 2.2] x_2^{\pm} &\leq [0.9, 1.1], \\ x_1^{\pm} \geq 0, \quad x_2^{\pm} &\geq 0. \end{aligned} \quad (33)$$

In stage one, suboptimal variables can be calculated using the GA nonlinear program solver engine of GASGOT:

$$x_1^s = 0.26497, \quad x_2^s = 0.37772, \quad f^s = 4.397. \quad (34)$$

In stage two,  $x_1^s$  and  $x_2^s$  are used to construct the objective functions expressed in (27), (28), (29), and (30) in order to determine the coefficients  $a_{ij}^{\pm+}$ ,  $b_i^{\pm+}$ ,  $c_j^{\pm+}$ , and  $d_j^{\pm+}$  and  $a_{ij}^{\pm-}$ ,  $b_i^{\pm-}$ ,  $c_j^{\pm-}$ , and  $d_j^{\pm-}$ . By solving (27), (28), (29), and (30), the coefficients are identified as follows:

$$\begin{aligned} a_{11}^{\pm+} &= 4.5, & a_{12}^{\pm+} &= 1.8, & a_{21}^{\pm+} &= 1.8, \\ b_1^{\pm+} &= 2.1, & b_2^{\pm+} &= 1.1, \\ c_1^{\pm+} &= 18, & c_2^{\pm+} &= 4, & d_1^{\pm+} &= 12, & d_2^{\pm+} &= 15, \\ a_{11}^{\pm-} &= 5.5, & a_{12}^{\pm-} &= 2.2, & a_{21}^{\pm-} &= 2.2, \\ b_1^{\pm-} &= 1.8, & b_2^{\pm-} &= 0.9, \\ c_1^{\pm-} &= 16, & c_2^{\pm-} &= 5, & d_1^{\pm-} &= 14, & d_2^{\pm-} &= 14. \end{aligned} \quad (35)$$

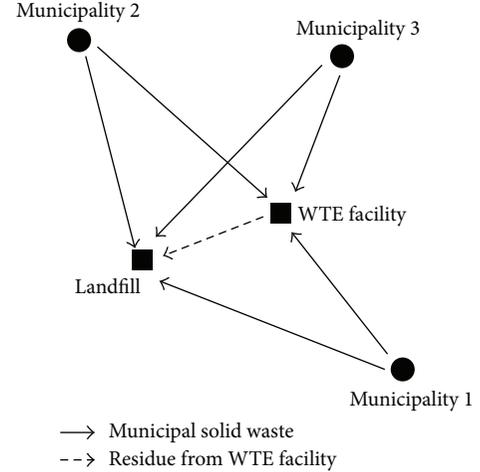


FIGURE 5: Diagram of municipalities and waste management facilities in the case study.

In stage three, the problems of (31) and (32) are generated as follows:

$$\begin{aligned} \text{Max } f^+ &= 18x_1^{\pm} - 12(x_1^{\pm})^2 - 4x_2^{\pm} + 15(x_2^{\pm})^2, \\ \text{s.t. } 4.5x_1^{\pm} + 1.8x_2^{\pm} &\leq 2.1, \\ x_1^{\pm} + 1.8x_2^{\pm} &\leq 1.1, \\ x_1^{\pm} \geq 0, \quad x_2^{\pm} &\geq 0, \\ \text{Max } f^- &= 16x_1^{\pm} - 14(x_1^{\pm})^2 - 5x_2^{\pm} + 14(x_2^{\pm})^2, \\ \text{s.t. } 5.5x_1^{\pm} + 2.2x_2^{\pm} &\leq 1.8, \\ x_1^{\pm} + 2.2x_2^{\pm} &\leq 0.9, \\ x_1^{\pm} \geq 0, \quad x_2^{\pm} &\geq 0. \end{aligned} \quad (36)$$

Solving the above two problems, the solution of this sample problem is  $f^{\pm} = [2.6371, 5.4234]$ ,  $x_1^{\pm} = [0.2441, 0.2857]$ , and  $x_2^{\pm} = [0.20792, 0.45239]$ . As a comparison, the solution given by [27] is  $f^{\pm} = [2.59, 5.42]$ ,  $x_1^{\pm} = [0.238, 0.286]$ , and  $x_2^{\pm} = [0.224, 0.452]$ .

To enhance the solution, the GAIQP engine can be reconfigured in stage three. For example, the maximum genetics generations number can be increased. However, this may not be necessary as the above generated solution is sufficiently satisfactory for many practical engineering problems.

#### 4. Case Study

To illustrate the proposed method, the problem of solid waste disposal planning presented in [25] has been recalculated using the GAIQP method. In this case study, the system involves three cities. As shown in Figure 5, the planning horizon is 15 years, which is divided equally into three periods. A landfill and an incinerator are available for the disposal of the municipal solid waste. The landfill has an

existing capacity of  $[2.05, 2.30] \times 10^6$  t, and the incinerator has a capacity of  $[500, 600]$  t/d. The incinerator generates residues of approximately 30% of the incoming waste streams, and its revenue from energy sale is  $\$[15, 25]$  per ton combusted. The waste generation rate of each city, operating costs of the facilities, and the waste transportation costs are summarized and shown in Table 2.

The objective of the optimization problem in this case study is to minimize the total costs by allocating waste flow between cities and facilities. In [25], this IQP model was formulated as follows:

$$\begin{aligned} \text{Min } f^\pm &= \sum_{i=1}^2 \sum_{j=1}^3 \sum_{k=1}^3 L_k (TR_{ijk}^\pm + OP_{ik}^\pm) x_{ijk}^\pm \\ &+ \sum_{j=1}^3 \sum_{k=1}^3 L_k [FE (FT_k^\pm + OP_{2k}^\pm) - RE_k^\pm] x_{2jk}^\pm \\ &\text{(where } TR_{ijk}^\pm = \alpha_{ijk}^\pm x_{ijk}^\pm + \beta_{ijk}^\pm, \\ &FT_{jk}^\pm = \sigma_k^\pm x_{2jk}^\pm FE + \delta_k^\pm), \\ \text{s.t. } &\sum_{j=1}^3 \sum_{k=1}^3 L_k (x_{1jk}^\pm + FE x_{2jk}^\pm) \\ &\leq TL^\pm \quad (\text{landfill capacity constraint}), \\ &\sum_{j=1}^3 x_{2jk}^\pm \leq TE^\pm \quad \forall k \text{ (incinerator} \\ &\quad \text{capacity constraint)}, \\ &\sum_{i=1}^2 x_{ijk}^\pm \geq WG_{jk}^\pm \quad \forall j, k \text{ (waste disposal} \\ &\quad \text{demand constraint)}, \\ &x_{ijk}^\pm \geq 0 \quad \forall i, j, k \text{ (nonnegativity constraint)}, \end{aligned} \tag{37}$$

where

$FE$  is residue flow rate from incinerator to landfill (it is 0.3 in this case),

$FT_k^\pm$  is transportation cost for residue from incinerator to landfill during period  $k$  ( $\$/t$ ),

$i$  is type of waste management facility ( $i = 1, 2$ , where  $i = 1$  for landfill, 2 for incinerator),

$j$  is city,  $j = 1, 2, 3$ ,

$k$  is time period,  $k = 1, 2, 3$ ,

$L_k$  is length of period  $k$ ,  $L_1 = L_2 = L_3 = 365 * 5$  (day),

$OP_{ik}^\pm$  is operating cost of facility  $i$  during period  $k$  ( $\$/t$ ),

$RE_k^\pm$  is revenue from incinerator during period  $k$  ( $\$/t$ ),

$RE_1^\pm = RE_2^\pm = RE_3^\pm = [15, 25]$ ,

$TE^\pm$  is capacity of incinerator (t/d),

$TL^\pm$  is capacity of landfill (t),

$TL_{ijk}^\pm$  is transportation cost for waste from city  $j$  to facility  $i$  during period  $k$  ( $\$/t$ ),

$WG_{jk}^\pm$  is waste generation rate in city  $j$  during period  $k$  (t/d),

$x_{ijk}^\pm$  is waste flow from city  $j$  to facility  $i$  during period  $k$  (t/d),

$\alpha_{ijk}^\pm$  is slope of transportation cost curve for waste from city  $j$  to facility  $i$ , during period  $k$ ,

$\beta_{ijk}^\pm$  is  $Y$ -intercept of transportation cost curve for waste from city  $j$  to facility  $i$  during period  $k$ ,

$\delta_k^\pm$  is  $Y$ -intercept of transportation cost curve for residue from incinerator to landfill during period  $k$ ,

$\sigma_k^\pm$  is Slope of transportation cost curve for residue from incinerator to landfill during period  $k$ .

This interval quadratic programming problem has 18 variables  $x_{ijk}$  and 13 constraints. The GAIQP method is applied to solve this problem according to the three stages presented in Section 3.2.

*Stage 1.* Based on the data shown in Table 2, the model formulated at stage 1 is as follows:

$$\begin{aligned} \text{min } f &= 365 \times 5 \\ &\times \{ [44.6, 64.4] x_{111} \\ &- [0.0123, 0.0163] x_{111}^2 \\ &+ [56.04, 81.34] x_{112} \\ &- [0.0135, 0.0179] x_{112}^2 \\ &+ [67.64, 103.48] x_{113} \\ &- [0.0148, 0.0197] x_{113}^2 \\ &+ [42.65, 61.9] x_{121} \\ &- [0.0106, 0.0142] x_{121}^2 \\ &+ [53.9, 78.56] x_{122} \\ &- [0.0117, 0.0156] x_{122}^2 \end{aligned}$$

TABLE 2: Waste generation rates and operating costs.

	Period 1 ( $k = 1$ )	Period 2 ( $k = 2$ )	Period 3 ( $k = 3$ )
Waste generation (t/d)			
$WG_{1k}^{\pm}$	[260, 340]	[310, 390]	[360, 440]
$WG_{2k}^{\pm}$	[160, 240]	[185, 265]	[210, 290]
$WG_{3k}^{\pm}$	[260, 340]	[260, 340]	[310, 390]
Operation cost (\$/t)			
$OP_{1k}^{\pm}$	[35, 45]	[40, 60]	[50, 80]
$OP_{2k}^{\pm}$	[55, 75]	[60, 85]	[65, 95]
City-to-landfill transportation cost (\$/t)			
$TR_{11k}^{-}$	$-0.0123x + 14.58$	$-0.0135x + 16.04$	$-0.0148x + 17.64$
$TR_{11k}^{+}$	$-0.0163x + 19.40$	$-0.0179x + 21.34$	$-0.0197x + 23.48$
$TR_{12k}^{-}$	$-0.0106x + 12.65$	$-0.0117x + 13.92$	$-0.0129x + 15.31$
$TR_{12k}^{+}$	$-0.0142x + 16.87$	$-0.0156x + 18.56$	$-0.0172x + 20.41$
$TR_{13k}^{-}$	$-0.0129x + 15.30$	$-0.0141x + 16.83$	$-0.0156x + 18.52$
$TR_{13k}^{+}$	$-0.0172x + 20.49$	$-0.0189x + 22.53$	$-0.0208x + 24.79$
City-to-incinerator transportation cost (\$/t)			
$TR_{21k}^{-}$	$-0.0097x + 11.57$	$-0.0107x + 12.73$	$-0.0118x + 14.00$
$TR_{21k}^{+}$	$-0.0130x + 15.42$	$-0.0143x + 16/97$	$-0.0157x + 18.66$
$TR_{22k}^{-}$	$-0.0102x + 12.17$	$-0.0113x + 13.39$	$-0.0124x + 14.73$
$TR_{22k}^{+}$	$-0.0136x + 16.15$	$-0.0149x + 17.76$	$-0.0164x + 19.54$
$TR_{23k}^{-}$	$-0.0089x + 10.60$	$-0.0098x + 11.67$	$-0.0108x + 12.83$
$TR_{23k}^{+}$	$-0.0118x + 14.10$	$-0.0130x + 15.51$	$-0.0143x + 17.06$
Incinerator-to-landfill transportation cost (\$/t)			
$FT_k^{-}$	$-0.0048x + 5.71$	$-0.0053x + 6.28$	$-0.0058x + 6.91$
$FT_k^{+}$	$-0.0064x + 7.62$	$-0.0070x + 8.38$	$-0.0077x + 9.33$

$$\begin{aligned}
& + [65.3, 100.4] x_{123} && + [66.2, 102.76] x_{222} \\
& - [0.0129, 0.0172] x_{123}^2 && - [0.013, 0.0172] x_{222}^2 \\
& + [45.3, 65.5] x_{131} && + [72.13, 114.54] x_{223} \\
& - [0.0129, 0.0172] x_{131}^2 && - [0.0143, 0.0186] x_{223}^2 + [60.3, 99.1] x_{231} \\
& + [56.83, 82.53] x_{132} && - [0.0102, 0.0137] x_{231}^2 \\
& - [0.0141, 0.0189] x_{132}^2 && + [65.41, 108.7] x_{232} \\
& + [58.52, 104.79] x_{133} && - [0.0104, 0.0148] x_{232}^2 \\
& - [0.0156, 0.0208] x_{133}^2 && + [71.38, 133.25] x_{233} \\
& + [60.6, 100.4] x_{211} && - [0.012, 0.0143] x_{233}^2 \}, \\
& - [0.0111, 0.015] x_{211}^2 + [64.2, 115.3] x_{212} && \text{s.t. } 365 \times 5 \\
& - [0.0122, 0.0164] x_{212}^2 && \times (x_{111} + 0.3x_{211} + x_{112} + 0.3x_{212} \\
& + [74.4, 126.73] x_{213} && + x_{113} + 0.3x_{213} + x_{121} + 0.3x_{221} \\
& - [0.0134, 0.018] x_{213}^2 && + x_{122} + 0.3x_{222} + x_{123} + 0.3x_{223} \\
& + [54.3, 103.44] x_{221} && + x_{131} + 0.3x_{231} + x_{132} + 0.3x_{232} \\
& - [0.0117, 0.0157] x_{221}^2 && + x_{123} + 0.3x_{233})
\end{aligned}$$

$$\begin{aligned}
&\leq [2.05 \times 10^6, 2.30 \times 10^6], \\
x_{211} + x_{221} + x_{231} &\leq [500, 600], \\
x_{212} + x_{222} + x_{232} &\leq [500, 600], \\
x_{213} + x_{223} + x_{233} &\leq [500, 600], \\
x_{111} + x_{211} &\geq [260, 340], \\
x_{112} + x_{212} &\geq [310, 390], \\
x_{113} + x_{213} &\geq [360, 440], \\
x_{121} + x_{221} &\geq [160, 240], \\
x_{122} + x_{222} &\geq [185, 265], \\
x_{131} + x_{231} &\geq [260, 340], \\
x_{132} + x_{232} &\geq [260, 340], \\
x_{133} + x_{233} &\geq [310, 390].
\end{aligned} \tag{38}$$

Considering interval numbers as random numbers, which can be determined between the interval numbers' lower and upper endpoints, the GAIQP method is applied to find a suboptimal solution  $f^s$ . With an initial population size of 500, after 2000 generations, one suboptimal solution has been found:  $f^s = 288 \times 10^6$ , and the concomitant variables are

$$\begin{aligned}
x_{111}^s &= 250, & x_{112}^s &= 315, & x_{113}^s &= 376, \\
x_{121}^s &= 3, & x_{122}^s &= 220, & x_{123}^s &= 57, \\
x_{131}^s &= 8, & x_{132}^s &= 10, & x_{133}^s &= 5, \\
x_{211}^s &= 45, & x_{212}^s &= 40, & x_{213}^s &= 10, \\
x_{221}^s &= 188, & x_{222}^s &= 12, & x_{223}^s &= 201, \\
x_{231}^s &= 278, & x_{232}^s &= 293, & x_{233}^s &= 322.
\end{aligned} \tag{39}$$

*Stage 2.* Substitute  $x_{ijk}^s$  into (38) to identify those coefficients

corresponding to  $f^+$  and  $f^-$ .

*Stage 3.* In this stage, the problem expressed in (37) has been converted into two subproblems:  $f^+$  and  $f^-$ .

The solution found by this method is  $f^\pm = [2.405 \times 10^8, 5.133 \times 10^8]$ . This is close to the result given by [25], which is, for comparison purposes,  $f^\pm = [2.39 \times 10^8, 5.14 \times 10^8]$ . This case study indicates that the GAIQP method can be configured to deal with large-scaled and complex engineering problems and give a satisfactory solution. If the parameters of the genetic algorithms of the nonlinear program solver engine of GASGOT are further tuned, a better solution can be generated.

## 5. Conclusions

Two genetic-algorithms-based methods have been proposed and applied for identifying an all-purpose optimization

solution for the inexact linear programming and Inexact quadratic programming problems. The two methods are called GAILP and GAIQP. The Genetic Algorithm Solver of MATLAB was the implementation environment of the proposed methods. Compared to the GAILP and GAIQP methods, the traditional problem-solving method has limitations due to the complexity involved in selecting the upper or lower bounds of variables and parameters when the subobjective functions are being constructed. The complexity arises due to the extensive computation and necessary assumptions and simplification. The solution procedures of the proposed GA-based optimization methods do not involve any such assumption or simplification, and the quality of the result is guaranteed. The GAIQP method has been applied to a case study that deals with municipal solid waste management taken from [25]. A comparison of the results shows that the proposed GA-based heuristic optimization approach is able to handle more complicated quadratic relationships involving uncertainty and provide better results.

The GA-based heuristic optimization approach is a flexible approach, which can be extended to find solutions for various types of operation programming scenarios. It can also be used as an all-purpose algorithm for economic optimizations. In the future, the methods of GAILP and GAIOP will be further developed for handling more complex inexact nonlinear problems.

## Acknowledgment

The financial support of the Canada Research Chair Program of Canada is gratefully acknowledged.

## References

- [1] L. E. Anderson, "A mathematical model for the optimization of a waste management system," Technical Report 68-1, Sanitary Engineering Research Laboratory, University of California, Berkeley, Calif, USA, 1968.
- [2] H. L. Christensen and G. F. Haddix, "A model for sanitary landfill management and design," *Computers and Operations Research*, vol. 1, no. 2, pp. 275–281, 1974.
- [3] L. A. Fuertes, J. F. Hundson, and D. H. Mark, "Solid Waste Management: equity trade-off models," *Journal of Urban Planning and Development*, vol. 100, pp. 155–171, 1974.
- [4] T. L. Jacobs and J. W. Everett, "Optimal scheduling of consecutive landfill operations with recycling," *Journal of Environmental Engineering*, vol. 118, no. 3, pp. 420–429, 1992.
- [5] L. Jenkins, "Parametric mixed integer programming—an application to solid waste management," *Management Science*, vol. 28, no. 11, pp. 1270–1284, 1982.
- [6] N.-B. Chang, C. G. Wen, Y. L. Chen, and Y. C. Yong, "A grey fuzzy multiobjective programming approach for the optimal planning of a reservoir watershed. Part B: application," *Water Research*, vol. 30, no. 10, pp. 2335–2340, 1996.
- [7] N.-B. Chang, C. A. Shoemaker, and R. E. Schuler, "Solid waste management system analysis with air pollution and leachate impact limitations," *Waste Management and Research*, vol. 14, no. 5, pp. 463–481, 1996.
- [8] S. Sushil and P. Vrat, "Waste management policy analysis and growth monitoring: an integrated approach to perspective

- planning," *International Journal of Systems Science*, vol. 20, no. 6, pp. 907–926, 1989.
- [9] N.-B. Chang, C. G. Wen, Y. L. Chen, and Y. C. Yong, "A grey fuzzy multiobjective programming approach for the optimal planning of a reservoir watershed. Part A: theoretical development," *Water Research*, vol. 30, no. 10, pp. 2329–2334, 1996.
- [10] M. Ekmekçiöğlü, T. Kaya, and C. Kahraman, "Fuzzy multicriteria disposal method and site selection for municipal solid waste," *Waste Management*, vol. 30, no. 8-9, pp. 1729–1736, 2010.
- [11] Y. Li and G. Huang, "Dynamic analysis for solid waste management systems: an inexact multistage integer programming approach," *Journal of the Air and Waste Management Association*, vol. 59, no. 3, pp. 279–292, 2009.
- [12] Y. M. Zhang, G. H. Huang, and L. He, "An inexact reverse logistics model for municipal solid waste management systems," *Journal of Environmental Management*, vol. 92, no. 3, pp. 522–530, 2011.
- [13] N.-B. Chang and S. F. Wang, "A fuzzy goal programming approach for the optimal planning of metropolitan solid waste management systems," *European Journal of Operational Research*, vol. 99, no. 2, pp. 303–321, 1997.
- [14] P. Guo and G. H. Huang, "Inexact fuzzy-stochastic mixed-integer programming approach for long-term planning of waste management—Part A: methodology," *Journal of Environmental Management*, vol. 91, no. 2, pp. 461–470, 2009.
- [15] Q. Tan, G. H. Huang, and Y. Cai, "A superiority-inferiority-based inexact fuzzy stochastic programming approach for solid waste management under uncertainty," *Environmental Modelling and Assessment*, vol. 15, no. 5, pp. 381–396, 2010.
- [16] Y. W. Lee, I. Bogardi, and J. Stansbury, "Fuzzy decision making in dredged-material management," *Journal of Environmental Engineering*, vol. 117, no. 5, pp. 614–630, 1991.
- [17] Y. Sun, Y. Li, and G. Huang, "Development of a fuzzy-queue-based interval linear programming model for municipal solid waste management," *Environmental Engineering Science*, vol. 27, no. 6, pp. 451–468, 2010.
- [18] G. Huang, B. W. Baetz, and G. G. Patry, "Grey quadratic programming and its application to municipal solid waste management planning under uncertainty," *Engineering Optimization*, vol. 23, pp. 201–223, 1994.
- [19] G. H. Huang, B. W. Baetz, and G. G. Patry, "Grey integer programming: an application to waste management planning under uncertainty," *European Journal of Operational Research*, vol. 83, no. 3, pp. 594–620, 1995.
- [20] G. Huang and R. D. Moore, "Grey linear programming, its solving approach, and its application," *International Journal of Systems Science*, vol. 24, no. 1, pp. 159–172, 1993.
- [21] G. H. Huang, B. W. Baetz, and G. G. Patry, "Grey dynamic programming for waste-management planning under uncertainty," *Journal of Urban Planning & Development*, vol. 120, no. 3, pp. 132–156, 1994.
- [22] G. H. Huang, B. W. Baetz, and G. G. Patry, "Grey dynamic programming for waste-management planning under uncertainty," *Journal of Urban Planning & Development*, vol. 120, no. 3, pp. 132–156, 1994.
- [23] G. Huang, B. W. Baetz, and G. G. Patry, "A grey fuzzy linear programming approach for municipal solid waste management planning under uncertainty," *Civil Engineering Systems*, vol. 10, no. 2, pp. 123–146, 1993.
- [24] G. Huang, B. W. Baetz, and G. G. Patry, "Waste flow allocation planning through a grey fuzzy quadratic programming approach," *Civil Engineering Systems*, vol. 11, pp. 209–243, 1994.
- [25] G. H. Huang, B. W. Baetz, and G. G. Patry, "Grey fuzzy integer programming: an application to regional waste management planning under uncertainty," *Socio-Economic Planning Sciences*, vol. 29, no. 1, pp. 17–38, 1995.
- [26] G. Huang, B. W. Baetz, and G. G. Patry, "A grey linear programming approach for municipal solid waste management planning under uncertainty," *Civil Engineering Systems*, vol. 9, no. 4, pp. 319–335, 1992.
- [27] M. J. Chen and G. H. Huang, "A derivative algorithm for inexact quadratic program—application to environmental decision-making under uncertainty," *European Journal of Operational Research*, vol. 128, no. 3, pp. 570–586, 2001.

## Research Article

# Composite Broadcasting and Ranging via a Satellite Dual-Frequency MPPSK System

Yu Yao, Lenan Wu, and Jiwu Wang

*School of Information Science and Engineering, Southeast University, Nanjing 210096, China*

Correspondence should be addressed to Yu Yao; 1057604987@qq.com

Received 18 May 2013; Revised 11 September 2013; Accepted 11 October 2013

Academic Editor: Saeed Balochian

Copyright © 2013 Yu Yao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Since digital video broadcasting via satellite (DVB-S) signals are “inefficient”, regarding the amount of information they convey on the bandwidth they occupy, a joint broadcasting and ranging system would constitute a unique platform for future digital video broadcasting satellite services effecting the essential tasks of satellite navigation system and direct to home (DTH) services, in terms of both spectrum efficiency and cost effectiveness. In this paper, the design of dual frequency M-ary position phase shift keying (MPPSK) system which is suitable for, respectively, performing both data transmission and range measurement is proposed. The approach is based on MPPSK modulation waveforms utilized in digital video broadcasting. In particular, requirements that allow for employing such signals for range measurements with high accuracy and high range are investigated. Also, the relationship between the frequency difference of dual frequency MPPSK system and range accuracy is discussed. Moreover, the selection of MPPSK modulation parameter for data rate and ranging is considered. In addition to theoretical considerations, the paper presents system simulations and measurement results of new systems, demonstrating the high spectral utilization of integrated broadcasting and ranging applications.

## 1. Introduction

Direct to home (DTH) services via satellite are particularly affected by power limitation, which weakens its ability of antinoise and antiinterference. Traditionally, power limitation is the main design objective rather than spectrum efficiency, so DVB-S system uses QPSK modulation [1]. However, the spectrum resource in the air has become more occupied, with rapid development of digital video and audio broadcasting. To achieve a very high spectrum efficiency without excessively penalizing the power efficiency, the modulation technique named M-ary position phase shift keying (MPPSK) was proposed, which is a kind of transmission technique with efficient-bandwidth and high data rate [2, 3].

Compared to EBPSK [4], MPPSK utilizes M-ary information symbols to directly control the positions of phase transition of sinusoidal carrier in each symbol cycle. These techniques offer a number of advantages, such as ultra-narrow bandwidth, very high transmission efficiency, and adjustable data rate. Different from QPSK, MPPSK preserves

a strong carrier within its RF spectra, because only a small portion of sine carrier is changed. Meanwhile, the receiver can extract the tiny modulation information to achieve demodulation while applying an extremely narrow pass-band filter named digital impacting filter (DIF). References [5, 6] explained the special mechanism of DIF elaborately. Continuous wave (CW) ranging has been demonstrated potential for providing precise detection results; however, they suffer from large range ambiguity [7]. Dual frequency CW ranging has a contradiction between the accuracy and range of measurement [8, 9].

In the paper, a dual frequency MPPSK system is creatively proposed to overcome the ranging ambiguity. At the same time, the system is used for the digital video broadcasting satellite services. Such a kind of platform would offer unique possibilities for novel system concepts and applications. Even more important, by using the dual frequency MPPSK waveform for both applications, the occupied spectrum would be used very efficiently and both applications could be operated, respectively, which would guarantee availability of both

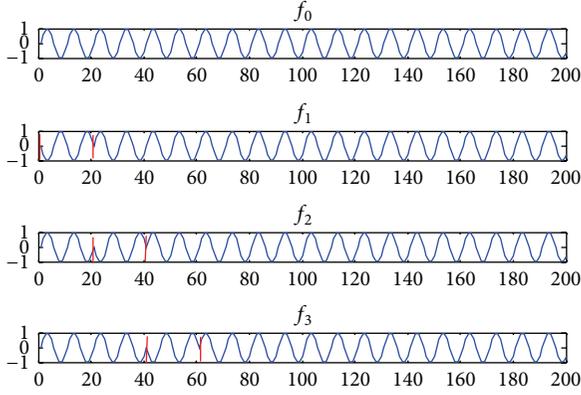


FIGURE 1: 4 PPSK modulated waveforms.

functions, and help to partially overcome the limitation of spectral resources. Such a system will provide two functions on a single hardware platform of MPPSK modems.

The rest of this paper is organized as follows: Section 2 introduces MPPSK modulation and demodulation, including the waveform character and the performance of impacting filters. Section 3 illustrates principle of dual frequency CW ranging. In Section 4, a block diagram of the dual frequency MPPSK broadcasting and receiving system is partly described, and also, two working modes are established. Some indicative simulation results and performance analysis are presented in Section 5. And finally, Section 6 gives the conclusion of the paper.

## 2. MPPSK Modem

**2.1. MPPSK Modulation.** The modulated MPPSK signals are defined by [2], and the simplified expression is presented in the paper as follows:

$$s_{i0}(t) = \begin{cases} \sin(2\pi f_0 t) & 0 \leq t < NT, \quad k = 0, \\ \begin{cases} \sin(2\pi f_0 t) & 0 \leq t \leq (k-1)KT, \\ -\sin(2\pi f_0 t) & (k-1)KT < t < kKT, \end{cases} & 1 \leq k \leq M-1, \\ \sin(2\pi f_0 t) & kKT \leq t < NT, \end{cases} \quad (1)$$

where  $f_0$  and  $T$  represent the carrier frequency and the carrier period, respectively.  $K$  and  $N$  stand for the number of the carrier period in each time slot and the number of the carrier period in each symbol, respectively.  $N/K$  means the slot number in each symbol, and  $m$  ( $m = 0, 1, \dots, M-1$ ) is  $M$ -ary ( $M \geq 2$ ) source symbol. The waveforms of 4PPSK modulation are illustrated as in Figure 1. The coefficient for the abscissa axis is the index of a certain sample point. Setting  $K = 2$ ,  $N = 20$ .

The modulation waveform for symbol “0” is sinusoidal wave as shown in Figure 1. Figure 1 also illustrates the modulation waveform for symbol “1” with the phase hopping during the first two carrier period (from 0 to 20), the next (from 20 to 40) is for symbol “2”, and last (from 40 to 60) is for symbol “3”.

MPPSK modulation generator is a new analog-digital-mixed type. Pulse train MPPSK modulator is illustrated by [3]. The modulating process of 4PPSK signal in carrier frequency  $f_1$  is shown in Figure 2.

The next step is to explain the processes more clearly, and the output of a sinewave oscillator with frequency  $f_1$  is divided into two branches: the upper is direct output with no phase shift, while the lower is phase reversed before output. The original data sequence to be transmitted is converted into a corresponding impulse chain firstly so as to control an electronic switch  $s$ . By (1), this chain always stays at low level (logic “0”) except for the beginning of bit “ $M$ ” ( $M > 0$ ) and during the interval of  $KT$  where the impulse appears and stays at high level (logic “1”). Controlled by this information impulse chain, the  $s$  is connected to  $s_0$  at low level, to  $s_1$  at high level.

**2.2. MPPSK Demodulation.** The impacting filter (IF) is a special digital infinite impulse response (IIR) filter, with the feature of “notch-frequency selection” in an extremely narrow pass-band [6]. It highlights the difference of the modulation waveform, which is helpful for demodulation, and simplifies the structure of the receiver greatly. At present, the IF is artificially designed. In the following real simulation, we assume the impacting filter formed by one pair of conjugate zeros and four pairs of adjacent conjugate poles. The expression and related parameter of the IF is given in [10].

Consider

$$H(z) = \frac{1 + b_1 \cdot z^{-1} + z^{-2}}{1 - \sum_{i=1}^{2n} a_i \cdot z^{-i}}, \quad (2)$$

where  $n$  is the pair number of the conjugate poles. In order to demodulate dual-frequency MPPSK signals, the zero parameters of the IF are selected as  $b_1 = -1.6181733185991785$ , and the pole parameters of the IF in this paper are selected as

$$\begin{aligned} a_1 &= -6.1150669443734404, & a_2 &= 17.593270854070781, \\ a_3 &= -30.66190141963812, & a_4 &= 35.258220132970798, \\ a_5 &= -27.343924194038685, & a_6 &= 13.991777506187015, \\ a_7 &= -4.3370740838799371, & a_8 &= 0.63250878596652416. \end{aligned} \quad (3)$$

The proposed filter has a very narrow bandwidth, and the IF would retain the signal characteristics and reduce the noise. When MPPSK modulated signals pass the impacting filter, the special impacting filter can transform the tiny waveform difference into amplitude impacting, and Figure 3 would illustrate this demodulation process.

The coefficient for the  $x$  axis is the index of a certain sample point. Set MPPSK Modulation parameter  $M = 4$ . Waveform of a source symbol sequence  $\{3, 2, 1, 0\}$  is shown in Figure 3, Figure 3 illustrates its 4PPSK waveform after the modulation, Figure 3 depicts the response of this filter to 4MPPSK modulated signals, the phase hopping can be converted into amplitude impacting, and its envelope is shown in Figure 3.

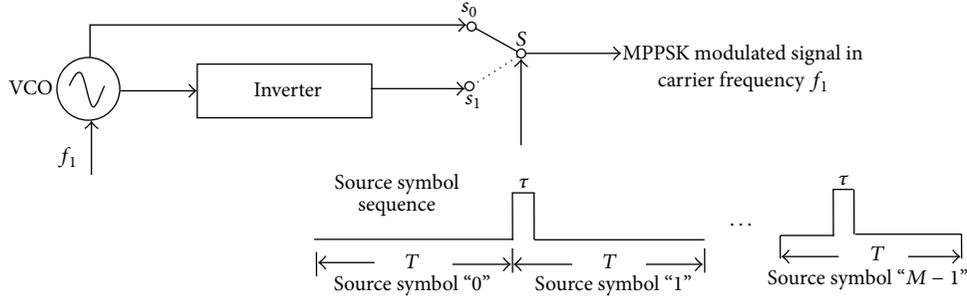


FIGURE 2: Block diagram of pulse train MPPSK modulator.

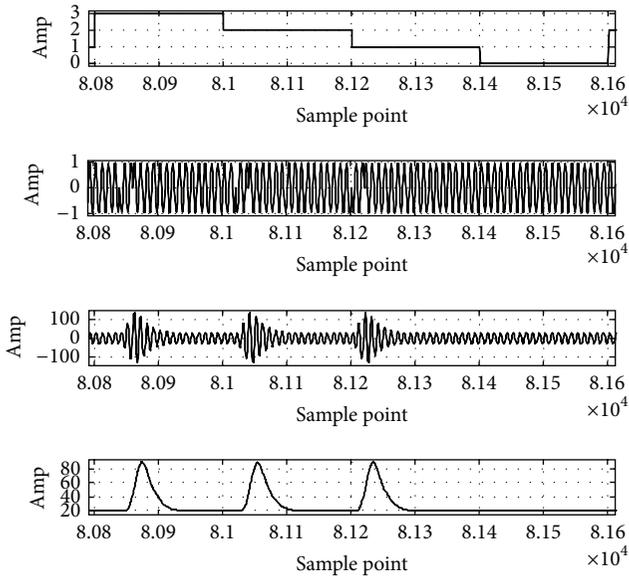


FIGURE 3: Block diagram of source, modulated signal, amplitude impacting, and envelope of amplitude impacting.

Nonzero symbols, that is, “1” to “ $M - 1$ ”, can be distinguished from the different positions of the peak of amplitude impacting. Figure 3 shows the waveform corresponding to four symbols, “3”, “2”, “1”, and “0.” The modulation waveform for symbol “0” is a sinusoidal. Therefore, no impacting is observed in its response. The peak of amplitude impacting (at about  $8.123 \times 10^4$  in Figure 3) corresponding to code “1” is close to the starting location (at about  $8.12 \times 10^4$  in Figure 3). The followed modulation waveform is for code “2” (at about  $8.106 \times 10^4$  in Figure 3), which is a little further from its own starting location (at about  $8.10 \times 10^4$  in Figure 3). And the last is for code “3” (at about  $8.088 \times 10^4$  in Figure 3), which is the furthest from its own starting location (at about  $8.08 \times 10^4$  in Figure 3). Obviously, a simple amplitude threshold detector and bit synchronization can perform the demodulation for MPPSK signals, which results in the simple receiver structure. The impacting filter can be digitally implemented, which is beneficial for chip integration.

### 3. Dual Frequency CW Ranging

Dual frequency CW ranging technique can compute very adequate fine range resolution [11, 12], without using frequency modulation. Consider a CW signal of single frequency first:

$$s_{t0}(t) = \sin(2\pi f_0 t), \quad (4)$$

where  $f_0$  is the carrier frequency and  $f_0 = 1/T$ . The receiver’s range  $R$  and is computed by measuring the time delay  $t_d$ , it takes CW signal to travel the path between transmitter and receiver. Since electromagnetic waves travel at the speed of light  $c$ , the received signal is

$$s_{r0}(t) = \sin[2\pi f_0(t - t_d)] = \sin(2\pi f_0 t - \varphi), \quad (5)$$

where the time delay  $\varphi = 2\pi f_0(R/c)$ .

Solving for  $R$ , we obtain

$$R = \frac{c\varphi}{2\pi f_0} = \frac{\lambda}{2\pi}\varphi. \quad (6)$$

Such a system with zero-intermediate frequency (ZIF) receiver is shown in Figure 4. PA stands for power amplifier and LNA represents low-noise amplifier;  $I/Q$  orthogonal channel is used in radar receiver for obtaining echo reflected signal phase and amplitude information. Clearly, the maximum unambiguous range occurs when  $\varphi$  is maximum; that is,  $\varphi = 2\pi$ . Consider a system with two CW signals, denoted by  $s_{t0}(t)$  and  $s_{t1}(t)$ , respectively. More precisely,

$$s_{t0}(t) = \sin(2\pi f_0 t + \phi_0), \quad s_{t1}(t) = \sin(2\pi f_1 t + \phi_1). \quad (7)$$

The received signals from transmitter are

$$\begin{aligned} s_{r0}(t) &= \sin(2\pi f_0 t + \phi_0 + \varphi_0), \\ s_{r1}(t) &= \sin(2\pi f_1 t + \phi_1 + \varphi_1), \end{aligned} \quad (8)$$

where  $\varphi_0 = 2\pi f_0 R/c$  and  $\varphi_1 = 2\pi f_1 R/c$ .

After heterodyning (mixing) with the carrier frequency, the phase difference between the two received signals is

$$\Delta\varphi = \frac{2\pi f_0 R}{c} \Delta f_{10}, \quad (9)$$

where  $\Delta\varphi = \varphi_1 - \varphi_0$  and  $\Delta f_{10} = f_1 - f_0$ .



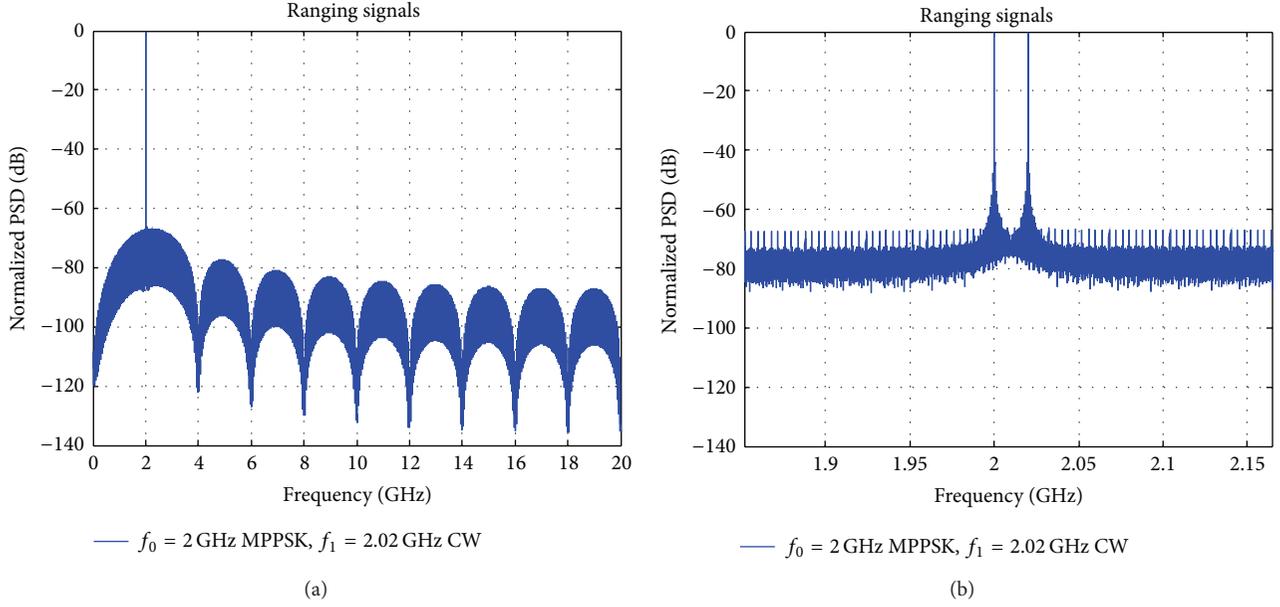


FIGURE 7: The PSD of MPPSK ranging signal.

$$\begin{aligned}
 & + \left( \frac{f_s}{M} \right)^2 \sum_{m=-\infty}^{\infty} \delta(f - mf_s) \\
 & \quad \times |G_0(mf_s) + G_1(mf_s) \\
 & \quad \quad + \cdots + G_{M-1}(mf_s)|^2 \\
 & + \frac{1}{4} \delta(f \pm f_1),
 \end{aligned} \tag{14}$$

where  $f_s = f_0/N$ ,  $G_0(f)$ ,  $G_1(f) \cdots G_{M-1}(f)$  is Fourier transform of MPPSK modulated waveforms corresponding to symbol “0”, “1”, ..., “ $M - 1$ ”, respectively.

**4.2. Broadcasting and Ranging Receiver.** Broadcast and ranging reception equipment is made by two kinds of receivers, and one is mainly composed of phase discriminator (PD), and the other is mainly composed of variable bit rate MPPSK (VBR-MPPSK) demodulation. As shown in Figure 6, receiver 1 and receiver 2 have been illustrated in Figure 4, and principle of MPPSK demodulation has been described in Section 2. The MPPSK signal in carrier frequency  $f_0$  is demodulated by VBR-MPPSK demodulation. The output data  $R_{10}$  can be converted to video after data processing. Signals both in carrier frequency  $f_0$  and in carrier frequency  $f_1$  are demodulated with PD. The output data  $R_{11}$  combined with  $R_{10}$  would be converted to distance value after data fusion.

Not only the employed waveform but also the general parameters have to be chosen according to the requirements derived from both applications. For broadcasting, digital video signals are demodulated by VBR-MPPSK demodulator alone. From (1), the variables  $M$ ,  $K$  and  $N$  form a parameter set to adjust the bandwidth, data rate and BER performance. Increasing value of  $M$  can lead to higher data because

more time slots are utilized. While taking multipath channel environment into consideration, large  $N$  is advantageous in mitigating multipath effect but pulls down the data rate and the efficiency of time slot. Therefore, small values of  $K/N$  results in sine-like waveform, which presents in solitude-peak low-sideband PSD appearance and occupies very limited bandwidth. So it is very important to trade off the parameters selection of MPPSK.

For ranging, theoretically, the maximum unambiguous range of MPPSK system must correspond to  $N$ , which is denoted by

$$R_{\max} = cT_0 = c \frac{N}{f_0}, \tag{15}$$

and  $K$  should be chosen according to  $\Delta f_{10}$  as

$$R_{10 \max} = \frac{c}{\Delta f_{10}} = c \frac{K}{f_0}. \tag{16}$$

So the maximum unambiguous ranging  $R_{11 \max}$  of dual frequency CW ranging system would be increased  $m$  times as follows:

$$R_{\max} = mR_{11 \max}. \tag{17}$$

Combining (15) and (16) yields

$$N \geq m \frac{f_0}{\Delta f_{10}}, \tag{18}$$

From Figure 6,  $R_{11}$  is high-precision measured value, and  $R_{10}$  is wide-range measured value. Combining the two measured values, the system would output a value of  $R$  with high-precision and wide-range; that is,

$$R = \left[ \frac{R_{10}}{R_{11 \max}} \right] R_{11 \max} + R_{11}, \tag{19}$$

where  $[ \ ]$  stands for round up.

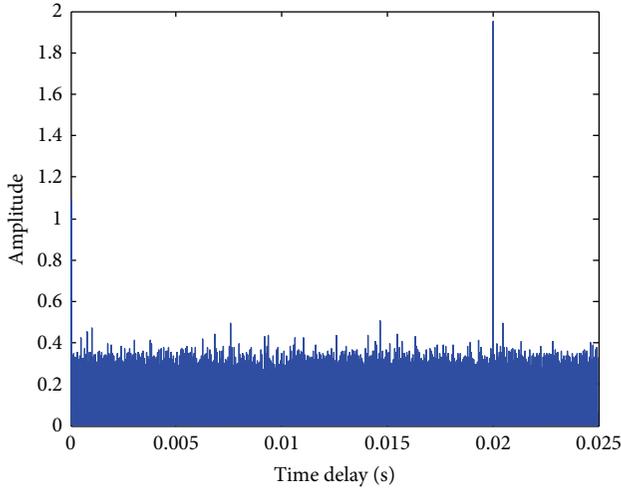


FIGURE 8: Simulation the rough range value.

TABLE 1: Simulation parameter for ranging.

Parameter	value
Carrier frequency:	$f_0 = 2 \text{ GHz}$ $f_1 = 2.02 \text{ GHz}$
Modulation:	$f_0$ : MPPSK, $f_1$ : CW
Modulation parameter:	$K = 100$ , $N = 5 \times 10^6$ , $M = 2$
Receiver distance:	600 km

Due to the increase in carrier frequency  $f_1$  of a CW signal, the dual frequency MPPSK signal becomes synthetic waveforms, that are suitable, respectively, for performing both data broadcasting and ranging. Compared with dual frequency CW ranging technique, the maximum unambiguous range is extended from  $c/\Delta f_{10}$  to  $N(c/f_0)$ .

## 5. Simulation Results

**5.1. Ranging Mode.** In this section, the performance of the proposed dual frequency MPPSK system is simulated. Firstly, we consider the dual frequency MPPSK system shown in Figures 5 and 6 in ranging mode. In Table 1, a summary of the most important parameters of the simulation model is provided.

As shown in Figure 7, the left subfigure is the global graph and the right one is the local enlarging graph for the carrier frequencies  $f_0$  and  $f_1$ . Obviously, the PSD of MPPSK ranging signal has a more narrow bandwidth with 99% power (or  $-60$  dB bandwidth). When  $K/N$  turns small, dual frequency MPPSK signal becomes more similar to pure sine waveform. Spectra of the ranging signal from system will be concentrated at carrier frequency  $f_0 = 2 \text{ GHz}$  and carrier frequency  $f_1 = 2.02 \text{ GHz}$ . Smaller portion of the sine carrier can be changed, and the line spectrum components in PSD of the MPPSK modulated signal become lower and even disappear. MPPSK tries to allocate as much power as possible to the carrier.

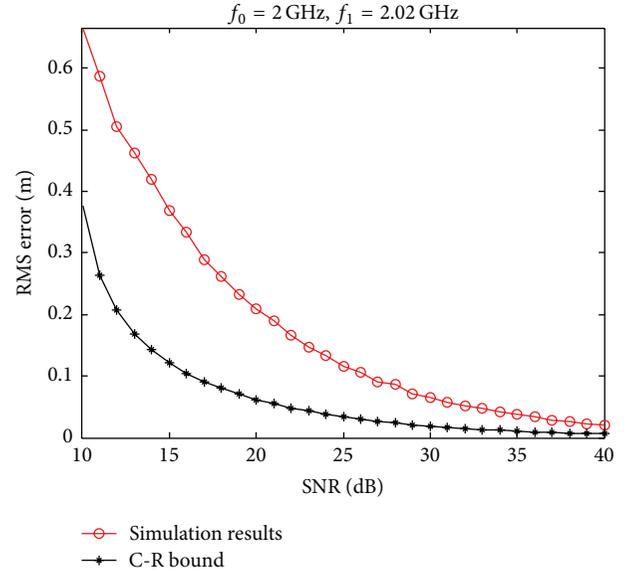


FIGURE 9: The simulation result and C-R bound.

The simulation experiment is in the AWGN channels (SNR = 2 dB), and as shown in Figure 8, the result is output of VBR-MPPSK demodulation. The rough range value can be computed by the time delay corresponding to amplitude impacting. The amplitude impacting is at the time delay of 1.9997 ms, and from (15), the rough range value  $R_{10}$  is 599 910 m.

From Figure 9, the simulation result and C-R bound just have approximately 0.3 m difference when SNR is 10 dB, with the increase of SNR, difference between simulation result and the theoretical result decreases, and RMS error shrinks towards equality in SNR = 40 dB.

**5.2. Broadcasting Mode.** We also consider the system shown in Figures 5 and 6 in broadcasting mode. In Table 2, a summary of the most important parameters of the simulation model is provided.

Figure 10 shows the PSD of QPSK and proposed MPPSK signal with the same modulation parameters. As shown in Figure 10, the left subfigure is the global graph of the PSD of 8PPSK and QPSK, and right one is the local enlarged graph nearing the carrier frequency  $f_0$ . Obviously, in MPPSK modulation, the line spectra, illustrating the periodic components of the modulated signal, decrease greatly, because of the random choosing of the  $M - 1$  positions. Therefore, MPPSK becomes more approximate to sine signal, the bandwidth with 99% power (or with  $-60$  dB bandwidth) decreases, and the spectra efficiency is improved greatly.

Figure 11 illustrates the SER in different modem, and the simulation shows that at  $\text{SER} = 10^{-4}$ , the SNR performance of the system with MPPSK modem may be improved by approximately 8 dB and 13 dB as compared with the performance of DVB-S with QPSK modulation and coherent demodulation and 16-QAM modulation and coherent demodulation, respectively.

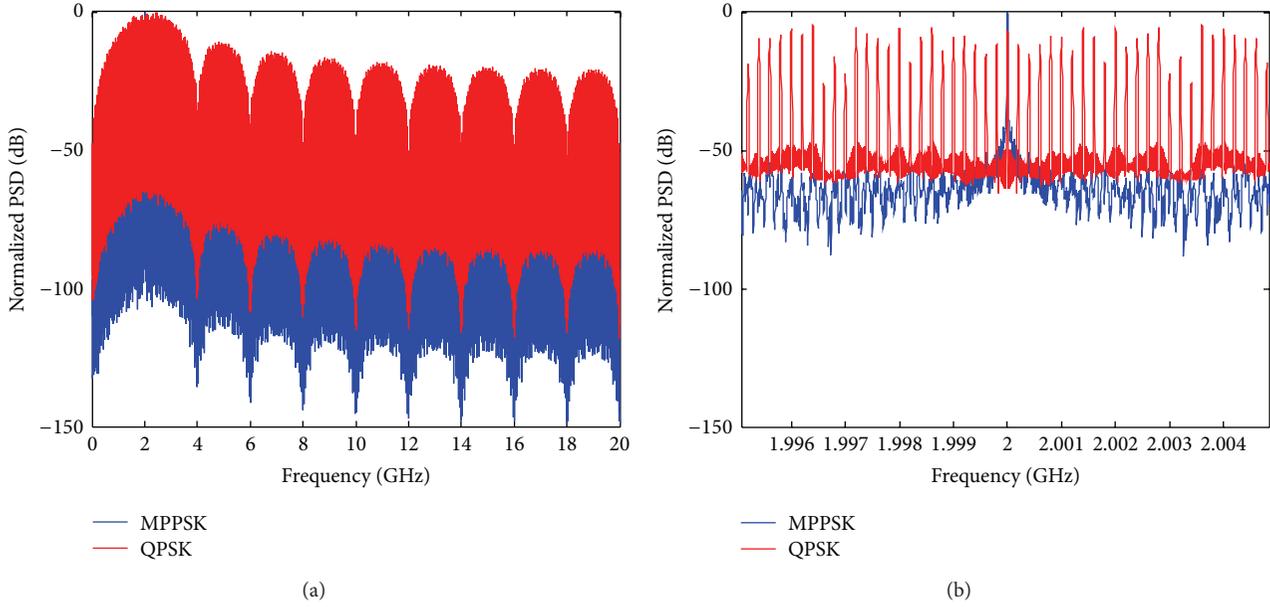


FIGURE 10: The PSD of 8PPSK and QPSK.

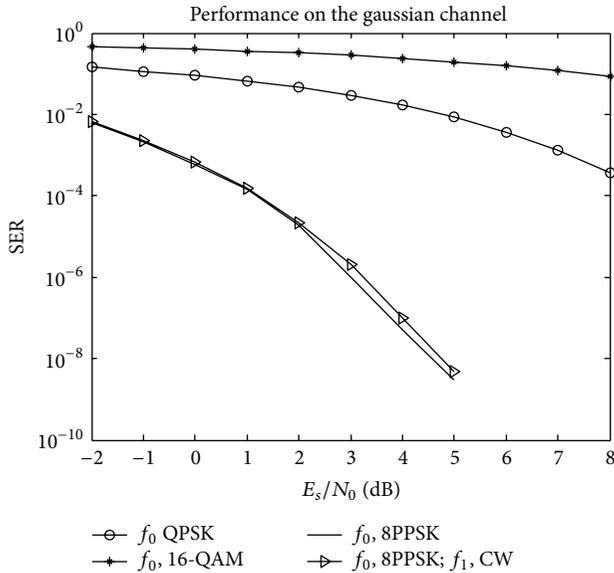


FIGURE 11: Comparative SER results of 8 PPSK, dual frequency 8 PPSK, QPSK and 16-QAM.

The result of the dual frequency MPPSK system and single frequency MPPSK system just have less than 1 dB difference in order to obtain the same SER performance, which is caused by the detection method and the selection of the decision threshold in the simulation. Simultaneously, research on the optimal modem method is underway; the system performance still has room for improvement.

### 6. Conclusions

The SER performance of the dual frequency MPPSK system is better than that of QPSK in such case, and its occupied

TABLE 2: Simulation parameters for broadcasting.

Parameter	value
Carrier frequency $f_0$	2 GHz
Modulation $f_0$	MPPSK
Modulation parameters:	$K = 2, N = 20, M = 8$
bit rate:	300 Mbps

bandwidth is much narrower than QPSK. QPSK tries to spread as much power as possible to the sidebands. On the opposite side, MPPSK allocates most power in the carrier to keep sideband energy emissions negligible, and dual frequency MPPSK system also provides an augmentation service, that is, the ranging function. After data integration between PD receiver and VBR-MPPSK demodulator, system would provide high precision ranging and extend maximum unambiguous range as well. The dual frequency MPPSK system is illustrated in block diagram, which is advantageous MPPSK signal generator and integrated receiver architecture, PD and impacting filter that essentially determine dual frequency signal demodulation are emphasized in the paper and correlation receiver distance simulation are made. The future work will continue with the research on dual frequency MPPSK system, including selection and combinations of other impacting filter and ranging algorithms. Different channel environment profile, as well as channel coding will be further considered.

### Acknowledgments

The authors thank all of the reviewers for their valuable comments, which have considerably helped in improving the overall quality of the work presented in the revised paper.

This work was supported by the National Natural Science Foundation of China (61271204).

## References

- [1] ETSI, "Digital broadcasting systems for television, sound and data service, framing structure, channel coding and modulation for 11/12 GHz satellite services," ETS 300 421, 1994.
- [2] C. Qi and L. Wu, "PLL demodulation technique for M-ray position phase shift keying," *Journal of Electronics*, vol. 26, no. 3, pp. 289–295, 2009.
- [3] P. K. Ying and L. N. Wu, "New scheme of MPPSK modem," *Journal of Southeast University*, vol. 42, no. 2, pp. 204–208, 2012.
- [4] L. Wu and M. Feng, "On BER performance of EBPSK-MODEM in AWGN channel," *Sensors*, vol. 10, no. 4, pp. 3824–3834, 2010.
- [5] J. Sun, W. Fang, and W. Xu, "A quantum-behaved particle swarm optimization with diversity-guided mutation for the design of two-dimensional IIR digital filters," *IEEE Transactions on Circuits and Systems II*, vol. 57, no. 2, pp. 141–145, 2010.
- [6] M. Feng, L. Wu, and P. Gao, "From special analogous crystal filters to digital impacting filters," *Digital Signal Processing*, vol. 22, no. 4, pp. 690–696, 2012.
- [7] P. van Genderen, "Recent advances in waveforms for radar, including those with communication capability," in *Proceeding of the European Radar Conference (EuRAD '09)*, pp. 318–325, Rome, Italy, October 2009.
- [8] C. Sturm and W. Wiesbeck, "Waveform design and signal processing aspects for fusion of wireless communications and radar sensing," *Proceedings of the IEEE*, vol. 99, no. 7, pp. 1236–1259, 2011.
- [9] Y. Zhang, M. Amin, and F. Ahmad, "A novel approach for multiple moving target localization using dual-frequency radars and time-frequency distributions," in *Proceeding of the 41st Asilomar Conference on Signals, Systems and Computers (ACSSC '07)*, pp. 1817–1821, Pacific Grove, Calif, USA, November 2007.
- [10] M. Feng, L. Wu, J. Ding, and C. Qi, "BER analysis and verification of EBPSK system in AWGN channel," *IEICE Transactions on Communications*, vol. E94-B, no. 3, pp. 806–809, 2011.
- [11] Weibel Inc, "Weibel RR-60034 ranging radar system".
- [12] Weibel, "Weibel azimuth & elevation monopulse doppler tracking radar systems," 1997–2002.
- [13] S. Peleg and B. Porat, "The Cramer-Rao lower bound for signals with constant amplitude and polynomial phase," *IEEE Transactions on Signal Processing*, vol. 39, no. 3, pp. 749–752, 1991.

## Research Article

# Research on ISFLA-Based Optimal Control Strategy for the Coordinated Charging of EV Battery Swap Station

Xueliang Huang,<sup>1,2</sup> Hao Qiang,<sup>1,3</sup> Qidong Zhang,<sup>1</sup> and Haijuan Li<sup>1</sup>

<sup>1</sup> School of Electrical Engineering, Southeast University, Nanjing 210096, China

<sup>2</sup> Jiangsu Key Lab of Smart Grid Technology and Equipment, Zhenjiang 212009, China

<sup>3</sup> School of Information Science and Engineering, Changzhou University, Changzhou 213164, China

Correspondence should be addressed to Xueliang Huang; [xlhuang@seu.edu.cn](mailto:xlhuang@seu.edu.cn)

Received 28 April 2013; Accepted 24 October 2013

Academic Editor: Yudong Zhang

Copyright © 2013 Xueliang Huang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

As an important component of the smart grid, electric vehicles (EVs) could be a good measure against energy shortages and environmental pollution. A main way of energy supply to EVs is to swap battery from the swap station. Based on the characteristics of EV battery swap station, the coordinated charging optimal control strategy is investigated to smooth the load fluctuation. Shuffled frog leaping algorithm (SFLA) is an optimization method inspired by the memetic evolution of a group of frogs when seeking food. An improved shuffled frog leaping algorithm (ISFLA) with the reflecting method to deal with the boundary constraint is proposed to obtain the solution of the optimal control strategy for coordinated charging. Based on the daily load of a certain area, the numerical simulations including the comparison of PSO and ISFLA are carried out and the results show that the presented ISFLA can effectively lower the peak-valley difference and smooth the load profile with the faster convergence rate and higher convergence precision.

## 1. Introduction

Over the past decades, many issues, such as energy shortages, serious environmental pollution, and global warming, have increasingly become worldwide concerns. EVs have emerged as a new traffic tool. Compared with internal combustion engine vehicles (ICEVs), which burn fossil fuels, EVs are driven by electricity. They demonstrate considerable advantages in solving the energy crisis and reducing the emissions of carbon dioxide, as well as in providing a means to drastically reduce the man-made pollution. More and more governments, car manufacturers, and energy companies are getting active in development and production of EVs [1].

With the large-scale introduction of EVs, the power grid will face a significant challenge. Many domestic and foreign scholars have carried out researches on the impact of EVs on power distribution system [2–7], which mainly focus on the coordinated charging of EVs but rarely involve the optimal control strategy for the coordinated charging of EV battery swap station. Here, the coordinated charging means that the batteries are controlled to orderly charge for achieving an

optimal objective such as minimizing the power losses or maximizing the grid load factor. Tian et al. established a mathematical model of dispatching strategy based on different objective functions which were solved by particle swarm optimization (PSO). The results show that the coordinated charging can lower the peak-valley difference and smooth the load profile [8].

The SFLA is a global optimization algorithm proposed by Eusuff et al. [9, 10]. It is a memetic metaheuristic that is based on the evolution of memes carried by interactive individuals and a global exchange of information among the frog population. The SFLA draws its formulation from two other search techniques: the local search of the “particle swarm optimization” technique and the competitiveness mixing of information of the “shuffled complex evolution” technique. It locates a global optimum by combining global information exchange and local search, simulating the process of a group of frogs’ population-based cooperative seeking food. The strategies of local search and global information exchange make frog leap out of local optimum towards the global optimum solution (food place) with high probability. The algorithm

has been developed to arrive at near-optimum solutions to complex and large-scale optimization problems which cannot be solved by gradient based mathematical programming techniques [11]. The SFLA has also been applied successfully to solving engineering problems such as traveling salesman problem (TSP) [12, 13], unit commitment (UC) problem [14–16], embedding virtual networks [17], speech emotion recognition [18], cognitive radio system [19], and other types of issues.

In this paper, based on the characteristics of the EV battery swap station, the ISFLA-based optimal control strategy for coordinated charging has been investigated. Through the discretization of the solution vector, an ISFLA based on the reflecting method to deal with the boundary constraint is proposed. The comparison of PSO and ISFLA shows that the presented ISFLA can lower the peak-valley difference and smooth the load profile with the faster convergence rate and higher convergence precision.

## 2. Shuffled Frog Leaping Algorithm

As a bioinspired optimization technique, the SFLA is a meta-heuristic optimization method, which imitates and models the behavior of frogs searching for food laid on discrete stones randomly located in a pond. In SFLA there is a population, which consists of a set of frogs (solutions). The set of frogs is partitioned into subsets referred to as memeplexes. Different memeplexes are considered as different cultures of frogs, and each memeplex performs a local search. Within each memeplex, the individual frogs hold ideas, which can be influenced by those of other frogs, and evolve through a process of memetic evolution. Frog leaping improves an individual's meme and enhances its performance towards the goal. After a predefined number of memetic evolution steps, ideas are passed among memeplexes in a shuffling process. The local search and the shuffling processes continue until the defined convergence criteria are satisfied.

The SFLA involves a population of possible solutions defined by a set of  $p$  randomly generated frogs denoted as  $Q = [X_1, X_2, \dots, X_p]^T$ , where  $X_i = (x_1, x_2, \dots, x_{iS})$  is the position of the  $i$ th frog in the  $S$  dimension search space. All the frogs are sorted in a descending order according to their fitness, and the population is divided into  $m$  memeplexes, each containing  $n$  frogs (i.e.,  $p = m \times n$ ), in such a way that the first frog goes to the first memeplex, the second frog goes to the second memeplex, the  $m$ th frog goes to the  $m$ th memeplex, and the  $(m + 1)$ th frog goes back to the first memeplex, and the process continues in this manner.

In each memeplex the frogs with the best and the worst fitness are represented by  $X_b$  and  $X_w$ , respectively. The best frog in the whole population is denoted by  $X_g$ . During memeplex evolution, the worst frog  $X_w$  leaps toward the best frog  $X_b$ , which is formulated as the following updating rule:

$$\begin{aligned} D &= \text{rand}(0, 1) \times (X_b - X_w), \\ X_{w.\text{new}} &= X_w + D, \quad -D_{\text{max}} \leq D \leq D_{\text{max}}, \end{aligned} \quad (1)$$

where  $D$  denotes the updated step size for frog leaping,  $\text{rand}(0,1)$  generates a random number between  $[0, 1]$ , and  $D_{\text{max}}$  is the maximum distance in one leaping.

If  $X_{w.\text{new}}$  has better fitness, the worst frog  $X_w$  will be replaced. Otherwise, the calculations in (1) are repeated with replacement of  $X_b$  by  $X_g$ . If no improvement occurs in this case, a new solution is randomly generated within the feasible space to replace the worst frog  $X_w$ . Then, the calculations continue for a specific number of iterations. After a pre-specified number of memetic evolutionary steps within each memeplex, to ensure global exploration, ideas passed within memeplexes are combined in the shuffling process. All the frogs are resorted, and the population is redivided into  $m$  memeplexes. The concurrently implemented local search and global shuffling continue alternatively until predefined convergence criteria are satisfied.

## 3. Charging Optimization Model for EV Battery Swap Station

Large-scale charging behavior of the EV will have a serious impact on the grid. Coordinated charging of the EV battery swap station can reduce the difference between the growing load peak and off-peak and save the costs of grid operation.

*3.1. Charging Power Model of the EV Battery Charger.* In the EV battery swap station, the typical strategy for battery charging is a two-stage method. The first stage has a constant current and limited pressure and the second stage has a constant pressure and limited current. Charging load power can be expressed as

$$P(t) = \begin{cases} I_{\text{max}}U(t), & 0 \leq t \leq T_C, \\ U_{\text{max}}I(t), & T_C < t \leq T_F, \end{cases} \quad (2)$$

where  $I_{\text{max}}$  and  $U_{\text{max}}$  denote the maximum charging current and voltage, respectively.  $T_C$  is the duration of charging with a constant current and  $T_F$  is the total charging duration. These parameters are constants determined by the battery type and charging characteristics.

In the first stage, to facilitate the modeling and analysis, the charging voltage is treated as the linear representation of  $U_{\text{max}}$ . And in the second stage, the charging current is exponentially declined with time [20]. Then (2) can be rewritten as

$$P(t) = \begin{cases} I_{\text{max}}U_{\text{max}} \left[ (1-k) \frac{t}{T_C} + k \right], & 0 \leq t \leq T_C, \\ U_{\text{max}}I_{\text{max}} e^{-\alpha(t-T_C)}, & T_C < t \leq T_F, \end{cases} \quad (3)$$

where  $\alpha$  and  $k$  are also constants determined by the battery type and charging characteristics.

In order to investigate the optimal control strategy for coordinated charging, the total charging duration  $T_F$  is divided equally into  $M$  sections, and the duration  $T_M$  of each section can be derived as  $T/M$ . According to the equal area

rule, the charging power of each section can be expressed with the average power:

$$\bar{P}_j = \frac{\int_{(j-1)T_M}^{jT_M} P(t) dt}{T_M}, \quad 1 \leq j \leq M. \quad (4)$$

Then an integer  $C$  is defined to satisfy the following expression:

$$CT_M \leq T_C \leq (C + 1)T_M. \quad (5)$$

Substituting (3) and (5) into (4) yields the multisection charging power model of battery charger:

$$\bar{P}_j = \begin{cases} U_{\max} I_{\max} \left[ \frac{(1-k)(2j-1)T_M}{2T_C} + k \right] & 0 < j \leq C \\ U_{\max} I_{\max} \left[ \frac{(1-k)(T_C^2 - C^2 T_M^2)}{2T_C} + k(T_C - CT_M) - \frac{e^{\alpha(T_C - CT_M - T_M)} - 1}{\alpha} \right] & C < j \leq C + 1 \\ U_{\max} I_{\max} \frac{e^{\alpha[T_C - (j-1)T_M]} - e^{\alpha(T_C - jT_M)}}{\alpha} & C + 1 < j \leq M. \end{cases} \quad (6)$$

Obviously, the accuracy of (6) is relative to  $M$ . A larger  $M$  implies the higher precision of the model with the greater amount of computation, and vice versa.

**3.2. Optimization Objective Function of the EV Battery Charging Power.** According to the “technical guide for electric vehicle battery-swap station” presented by the State Grid Corporation of China (SGCC), a single EV battery charger is used to charge a single battery box. Inside the battery box, the battery pack consists of a plurality of battery cells. The battery mentioned in this paper refers to the battery pack. There are two types of batteries in the EV battery swap station, namely, the regulated battery connected to the EV battery charger and participating in the grid charging optimization by controlling its charging start time and the full charged reserve battery disconnected to the charger and used when not meeting the demand for swapping battery. Only the regulated-battery is considered in this paper.

In order to investigate the optimal control strategy for coordinated charging of the EV battery swap station, some assumptions are made as follows.

(1) All batteries are of the same type. The initial-charging SOC of each battery is the same and set to 20%, and the end-charging SOC is also the same and set to 100%.

(2) Corresponding to the above multisection charging power model of the battery charger, one optimization period  $T_O$  is also divided into multisections ( $M_O$ ) with each section having the same duration  $T_M$ . And  $M_O = T_O \cdot M/T$ . Then

in one optimal period  $T_O$ , the number  $N_b$  of batteries needed for charging can be expressed as

$$N_b = \sum_{j=1}^{M_O} b_j, \quad 1 \leq j \leq M_O, \quad (7)$$

where  $b_j$  is the predicted number of batteries swapped into the station during the  $j$ th section. For optimal control, it is necessary to number these batteries needed for optimal charging. From the  $b_0$  batteries, uncharged and reserved in the swap station at the beginning of the  $T_O$ , to the batteries swapped into the station during the  $T_O$ , the batteries are numbered consecutively until  $N_b$ . The other batteries swapped into the station of the  $T_O$  are not indexed again and charged during the next optimization period.

(3) For the battery  $i$ ,  $t_{\text{in}}(i)$  and  $t_{\text{out}}(i)$  denote its swap-in time and swap-out time, respectively:

$t_{\text{in}}(i) = 0$ : battery  $i$  is one of the batteries  $b_0$ ;

$t_{\text{in}}(i) = t_j$ : battery  $i$  is swapped in during the  $j$ th optimization section;

$t_{\text{out}}(i) = t_k$ : battery  $i$  is swapped out during the  $k$ th optimization section;

$t_{\text{out}}(i) = M_O + 1$ : battery  $i$  is remained in the station.

The battery can be swapped into and swapped out of the station at any time in one section. To facilitate the optimal control, we suppose that the battery could be charged from the section next to that of swapping in and be swapped out of the station from the section next to that of full charged. Then the start-charging time  $T_S(i)$  of the battery  $i$  is between  $t_{\text{in}}(i) + 1$  and  $t_{\text{out}}(i) - M$ :

$$t_{\text{in}}(i) + 1 \leq T_S(i) \leq t_{\text{out}}(i) - M. \quad (8)$$

To ensure that all batteries are full charged at the end of the  $T_O$ , the swapped-in batteries during the last  $M$  sections of the  $T_O$  are no longer charged. Then the start-charging time  $T_S(N_b)$  of the battery  $N_b$  would meet

$$b_0 + \sum_{j=1}^{T_S(N_b)-1} b_j \leq N_b \leq b_0 + \sum_{j=1}^{T_S(N_b)} b_j, \quad (9)$$

$$T_S(N_b) \leq M_O - M.$$

If there are  $i_{f_0}$  full charged batteries at the beginning of the  $T_O$  and all swapped out in the  $t_f$  section, the new full charged batteries would be swapped out from the  $t_f$  section. Then

$$\sum_{j=1}^{t_f-1} b_j \leq i_{f_0} \leq \sum_{j=1}^{t_f} b_j, \quad t_f \geq M + 1. \quad (10)$$

(4) The battery is the core of EV and quite expensive. Many times of start and stop charging would greatly reduce its life, in which case the battery charging continuity constraint is introduced.

Based on assumptions (1) and (4), the charging duration of all batteries is the same. In this paper, the minimum variance of the load profile including the charging loads generated by the EV battery swap station is the optimization objective and is expressed as

$$\min F = \frac{\min \left[ \sum_{j=1}^{M_O} \left( P_{d,j} + \sum_{i=1}^N C_{i,j} P_{i,j} - P_{avr} \right)^2 \right]}{M_O}$$

$$\text{S.T.} \begin{cases} b_0 + \sum_{j=1}^{T_S(N_b)-1} b_j \leq N_b \leq b_0 + \sum_{j=1}^{T_S(N_b)} b_j \\ T_S(N_b) \leq M_O - M \\ \sum_{j=1}^{t_f-1} b_j \leq i_{f0} \leq \sum_{j=1}^{t_f} b_j \\ t_f \geq M + 1 \\ t_{in}(i) + 1 \leq T_S(i) \leq t_{out}(i) - M, \end{cases} \quad (11)$$

where

$$P_{avr} = \frac{\sum_{j=1}^{M_O} \left( P_{d,j} + \sum_{i=1}^N C_{i,j} P_{i,j} \right)}{M_O}, \quad (12)$$

$P_{d,j}$ : the predicted daily load power of the  $j$ th section excluding the charging load generated by EV swap station;

$N$ : the total number of EV battery chargers;

$C_{i,j}$ : the charging decision variable of the  $i$ th battery during the  $j$ th section: "1" means charging and "0" is discharging. Based on assumption (4),

$$C_{i,j} = \begin{cases} 1 & T_S(i) \leq j \leq T_S(i) + M - 1, \\ 0 & \text{else;} \end{cases} \quad (13)$$

$P_{i,j}$ : the charging load of the  $i$ th battery during the  $j$ th section.

#### 4. ISFLA-Based Optimal Control Strategy for Coordinated Charging

There are  $N$  chargers controlled by the control center of the EV battery swap station. From (11), the optimal control of the battery coordinated charging is a multivariable, nonlinear, and discrete integer optimization problem. Using the reflecting method to deal with the boundary constraint, an improved shuffled frog leaping algorithm (ISFLA) is proposed to achieve the optimal control strategy for coordinated charging of EV battery swap station.

**4.1. Definition of Frog Position.** In this paper, the position of the virtual frog consists of the start-charging time of every battery needed for coordinated charging, so the dimension

of solution space is equal to the number  $N_b$  of the regulated battery. The  $i$ th frog can be expressed as

$$X_i = (T_{is}(1), T_{is}(2), \dots, T_{is}(N_b)). \quad (14)$$

Based on the assumptions,

$$T_{is}(h) \in \begin{cases} [1, M_O - M] & 1 \leq h \leq b_0, \\ [K + 2, M_O - M] & \sum_{j=0}^K b_j + 1 \leq h \leq \sum_{j=0}^{K+1} b_j, \end{cases} \quad (15)$$

where  $0 \leq K \leq M_O - M - 2$ .

**4.2. Inner-Memplex Evolution.**  $T_{is}$  is an integer, and the position of the frog with the worst fitness is adjusted as the following modified rule:

$$d_j = \text{rand}(0, 1) \times (T_{bs}(j) - T_{ws}(j)), \quad 0 \leq j \leq N_b, \quad (16)$$

$$T_{w,\text{new}}(j) = T_{ws}(j) + [d_j], \quad -d_{j,\text{max}} \leq d_j \leq d_{j,\text{max}},$$

where  $[d_j]$  means an integer rounded to  $d_j$ .

**4.3. Boundary Constraint.** Dealing with the boundary constraint may affect the performance of the algorithm. In general, there are three basic approaches, namely, absorbing, reflecting, and damping. Using the reflecting method, when the updating position of the frog after  $t$  iterations is beyond a certain boundary, the frog would return within the boundary with the original speed but in the opposite direction. In this paper, the reflecting method is chosen to deal with the boundary constraint and the position evolution of the frog obeys the following rule.

When  $d_j > 0$ ,

$$\begin{aligned} T_{ws,\text{new}}(j) &= \begin{cases} T_{ws}(j) - [d_j] & T_{ws}(j) = M_O - M, \\ \min(T_{ws}(j) + [d_j], M_O - M) & T_{ws}(j) < M_O - M; \end{cases} \end{aligned} \quad (17)$$

when  $d_j < 0$ ,

$$\begin{aligned} T_{ws,\text{new}}(j) &= \begin{cases} T_{ws}(j) - [d_j] & T_{ws}(j) = t_{in}(j) + 1, \\ \max(T_{ws}(j) + [d_j], t_{in}(j) + 1) & T_{ws}(j) > t_{in}(j) + 1. \end{cases} \end{aligned} \quad (18)$$

**4.4. Steps of the ISFLA.** The overall process of the ISFLA can be described in the following main steps.

**Step 0** (initialize the parameters). Set the parameters of  $p$ ,  $n$ ,  $ite\_m$ , and  $ite\_p$ .  $p$  is number of frogs in the whole population.  $n$  is the number of frogs in each memplex.  $ite\_m$  is the maximum number of iterations for each memplex.  $ite\_p$  is the maximum number of shuffling iterations for whole population. Therefore,  $m = p/n$  is the number of memplexes.

TABLE 1: Basic parameters of battery.

Type	$U_{\max}$ (V)	$I_{\max}$ (A)	Constant current charging duration (min)	Constant voltage charging duration (min)
Li-Ion	58.8	120	90	150

*Step 1 (initialize the population).* The initial population is formed by  $p$  randomly generated frogs:  $X_1, X_2, \dots, X_p$ . Compute the fitness value  $F(X_i)$  according to (11) for each frog  $X_i$ .

*Step 2 (shuffle the frogs).* Firstly sort the  $p$  frogs in order of decreasing fitness, and then partition all the  $p$  frogs into  $m$  memplexes. The first frog goes to the first memplex, the second frog goes to the second memplex, the  $m$ th frog goes to the  $m$ th memplex, the  $(m+1)$ th frog goes back to the first memplex, and at last the  $p$ th frog goes to the  $m$ th memplex.

*Step 3 (memetic evolution).* Repeat the following operations for  $ite\_m$  times: firstly find the worst frog  $X_w$  and the best frog  $X_b$  in each memplex, then update  $X_w$  according to (16)–(18), and yield  $X_{w\_new}$ . If  $X_{w\_new}$  has a better fitness, replace  $X_w$  by  $X_{w\_new}$ . Otherwise, repeat the updated strategy with  $X_g$  replacing  $X_b$ . If this still cannot produce a better solution, replace the  $X_w$  by a randomly generated frog.

*Step 4 (check convergence).* If the convergence criteria are met, stop and output the best frog  $X_g$  in the whole population. Otherwise, return to Step 2.

In this ISFLA, the convergence criteria are defined as the maximum number of shuffling iterations for whole population ( $ite\_p$ ).

## 5. Numerical Simulation

In this paper, the optimization objective is one area's daily load profile which includes the charging loads generated by the EV battery swap station. The optimization period  $T_O$  is set to a day and divided into 24 sections with the duration  $T_M$  of one hour.

*5.1. Parameters of EV Battery.* In the EV market, there are different battery types such as NiMH, Lead Acid, and Li-Ion. And the market share of Li-Ion with its technical advantages has increased annually. The “E6 pioneer” EV developed by BYD Co., Ltd., has been configured a Li-Ion battery with the energy density of 100 W·h per kilogram (i.e., a battery of 600 kg can store power of 60 kW·h for each charging). Its parameters are shown in Table 1.

In consideration of the charging continuity constraint and based on the battery parameters, the average charging power of each section in the multisection charging model can be calculated and shown in Table 2.

*5.2. Area Daily Load.* A certain area daily load power before regulation can be predicted and shown in Table 3.

TABLE 2: Average charging power of each section.

Section	1	2	3	4
Charging power (kW)	6.45	6.18	2.52	0.85

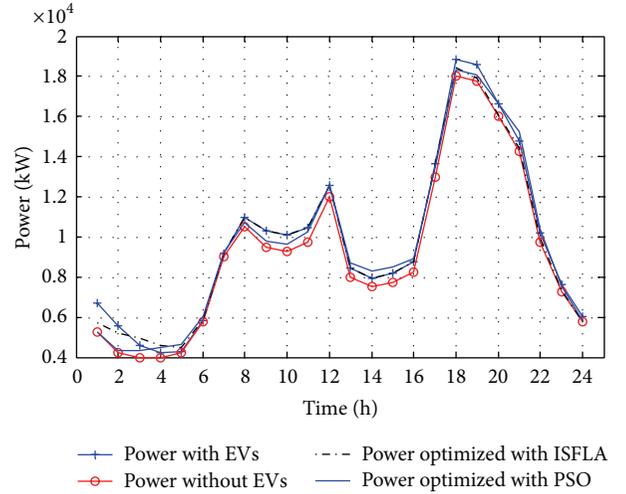


FIGURE 1: Area load profile under different conditions.

*5.3. Parameters of the EV Battery Swap Station.* At the beginning of one optimization period, there are 600 batteries in the swap station, including 350 full charged batteries, 200 unfull charged batteries, and 50 reserved batteries. All batteries except the reserved batteries participate in the optimal control of coordinated charging. By predicting the demand of EV owners for swapping batteries during each optimization section, the number  $b_j$  can be derived and shown in Table 4.

*5.4. Simulation of ISFLA.* The main parameters of ISFLA are set and shown in Table 5. Based on ISFLA and PSO, the area load profiles are compared and shown in Figure 1.

In Figure 1, ISFLA and PSO can lower the load peak-valley difference and smooth the load profile. But the convergence characteristics of two algorithms in the numerical simulation differ greatly and are shown in Figure 2.

Clearly, the convergence rate of ISFLA is much faster than that of PSO and the minimum variance of the load profile generated by ISFLA is less than that generated by PSO. Under different conditions, the variances of the load profile are shown in Table 6.

With ISFLA, the distribution of batteries according to the start-charging time in the optimization period  $T_O$  is shown in Figure 3.

In Figure 3, during some optimization sections, there is a very small or no difference between the number of starting-charging batteries and that of those swapped into the station. This can explain that the optimized load profile with ISFLA and the load profile with uncoordinated charging almost coincide in some sections in Figure 1.

For a fixed population  $p$ , the different number  $m$  of partitioned memplexes will affect the convergence rate and the global optimal solution. Some simulations are made of

TABLE 3: Certain area daily load before regulation.

Time (h)	Power (MW)						
1	5.25	7	9	13	8	19	17.75
2	4.25	8	10.5	14	7.5	20	16
3	4	9	9.5	15	7.75	21	14.25
4	4	10	9.25	16	8.25	22	9.75
5	4.25	11	9.75	17	13	23	7.25
6	5.75	12	12	18	18	24	5.75

TABLE 4: Predicted number of batteries swapped in during each optimization section.

Time (h)	1	2	3	4	5	6	7	8	9	10	11	12
Number	3	3	1	2	10	18	53	68	46	31	17	24
Time (h)	13	14	15	16	17	18	19	20	21	22	23	24
Number	30	30	37	50	67	42	28	28	25	18	13	6

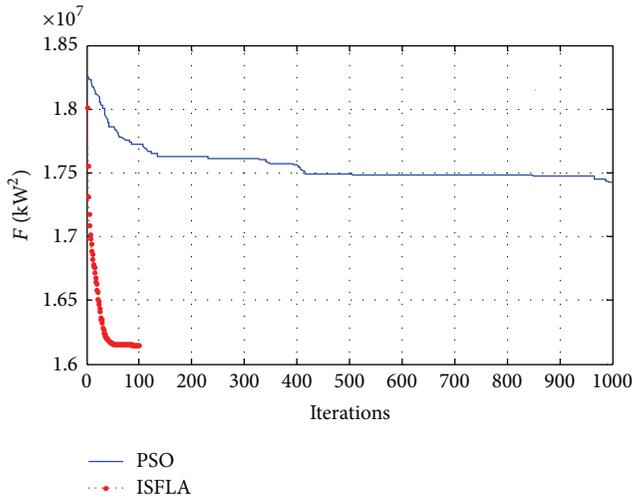


FIGURE 2: Convergence characteristics of ISFA and PSO.

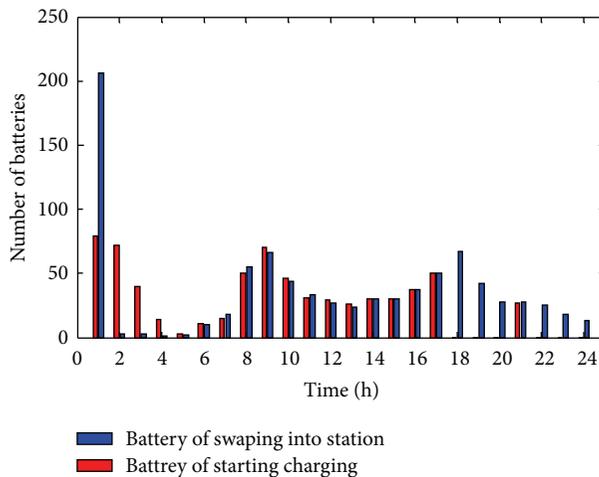


FIGURE 3: Distribution of batteries starting charging.

TABLE 5: Parameters of ISFLA.

Parameter	$p$	$m$	$n$	$ite\_m$	$ite\_p$
Value	50	10	5	5	100

TABLE 6: Variances under different conditions.

Condition	Uncoordinated charging	Coordinated charging	
		ISFLA	PSO
Variance ( $\text{kW}^2$ )	1.836	1.615	1.737
Result	—	Declining 12.04%	Declining 5.40%

TABLE 7: Global convergence value.

Memeplex	5	10	20	40
$F (\text{kW}^2) \times 10^7$	1.616	1.615	1.6139	1.6137

ISFLA with the parameters of a fixed  $p$  (200) and different  $m$  and the results are shown in Figure 4 and Table 7.

In Figure 4 and Table 7, the greater the number of memeplex is, the faster the algorithm converges with a higher convergence performance. However, the relationship between them is not linear, and how to choose an appropriate memeplex considering the convergence rate and global convergence value is a main goal of our further work.

## 6. Conclusion

As a main way of energy supply to EVs, the optimal control strategy for coordinated charging of the swap station is very important in smoothing the load profile. Based on the characteristics of the EV battery swap station, a multisection charging power model of battery is presented and an ISFLA in reflecting method to deal with the boundary constraint is proposed to achieve coordinated charging of batteries. In numerical simulations, the comparison of PSO and ISFLA is made, and the results show that the presented ISFLA can effectively lower the peak-valley difference and smooth

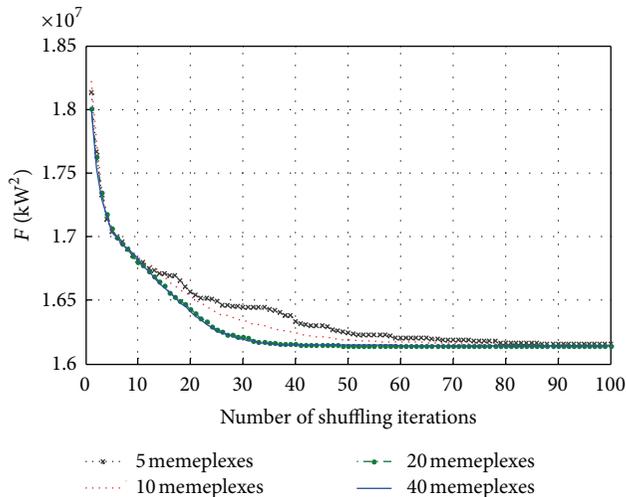


FIGURE 4: Iterations of ISFLA with different memplexes.

the load profile with the faster convergence rate and higher convergence precision.

## Acknowledgments

This work was supported by the National High-Tech Research & Development Program of China (“863” Program) (Grant no. 2012AA050210), the National Natural Science Foundation of China (Grant no. 51177011), and Provincial Science and Technology Supporting Program (Grant no. BE2011174).

## References

- [1] Q. Q. Chen and L. Q. Sun, “Present statue and future trends of electric vehicles,” *Science & Technology Review*, vol. 23, no. 4, pp. 24–28, 2011 (Chinese).
- [2] K. Schneider, C. Gerkenmeyer, M. Kintner-Meyer, and R. Fletcher, “Impact assessment of plug-in hybrid vehicles on pacific northwest distribution systems,” in *Proceedings of the IEEE Power and Energy Society General Meeting: Conversion and Delivery of Electrical Energy in the 21st Century*, pp. 1–6, July 2008.
- [3] S. Acha, T. C. Green, and N. Shah, “Impacts of plug-in hybrid vehicles and combined heat and power technologies on electric and gas distribution network losses,” in *Proceedings of the 1st IEEE-PES/IAS Conference on Sustainable Alternative Energy (SAE '09)*, pp. 1–7, September 2009.
- [4] W. Su and M.-Y. Chow, “Performance evaluation of an EDA-based large-scale plug-in hybrid electric vehicle charging algorithm,” *IEEE Transactions on Smart Grid*, vol. 3, no. 1, pp. 308–315, 2012.
- [5] K. Clement-Nyns, E. Haesen, and J. Driesen, “The impact of charging plug-in hybrid electric vehicles on a residential distribution grid,” *IEEE Transactions on Power Systems*, vol. 25, no. 1, pp. 371–380, 2010.
- [6] F. Lanati, M. Benini, and A. Gelmini, “Impact of the penetration of electric vehicles on the Italian power system: a 2030 scenario,” in *Proceedings of the IEEE PES General Meeting: The Electrification of Transportation and the Grid of the Future*, pp. 1–8, July 2011.
- [7] R. J. Bessa, M. A. Matos, F. J. Soares, and J. A. P. Lopes, “Optimized bidding of a EV aggregation agent in the electricity market,” *IEEE Transactions on Smart Grid*, vol. 3, no. 1, pp. 443–452, 2012.
- [8] W. Q. Tian, J. H. He, J. C. Jiang, L. Y. Niu, and X. I. Wand, “Research on dispatching strategy for coordinated charging of electric vehicle battery swapping station,” *Power System Protection and Control*, vol. 40, no. 21, pp. 114–119, 2012.
- [9] M. Eusuff, K. Lansey, and F. Pasha, “Shuffled frog-leaping algorithm: a memetic meta-heuristic for discrete optimization,” *Engineering Optimization*, vol. 38, no. 2, pp. 129–154, 2006.
- [10] M. M. Eusuff and K. E. Lansey, “Optimization of water distribution network design using the shuffled frog leaping algorithm,” *Journal of Water Resources Planning and Management*, vol. 129, no. 3, pp. 210–225, 2003.
- [11] X. Pan, “The study on the shuffled frog leap algorithm and its application,” *Advances in Information Sciences and Service Sciences*, vol. 4, no. 1, pp. 215–223, 2012.
- [12] X.-H. Luo, Y. Yang, and X. Li, “Modified shuffled frog-leaping algorithm to solve traveling salesman problem,” *Journal on Communications*, vol. 30, no. 7, pp. 130–135, 2009 (Chinese).
- [13] J.-P. Luo and X. Li, “Improved shuffled frog leaping algorithm for solving TSP,” *Journal of Shenzhen University Science and Engineering*, vol. 27, no. 2, pp. 173–179, 2010.
- [14] J. Ebrahimi, S. H. Hosseinian, and G. B. Gharehpetian, “Unit commitment problem solution using shuffled frog leaping algorithm,” *IEEE Transactions on Power Systems*, vol. 26, no. 2, pp. 573–581, 2011.
- [15] J. M. Anita and I. J. Raglend, “Solution of unit commitment problem using shuffled frog leaping algorithm,” in *Proceedings of the International Conference on Computing, Electronics and Electrical Technologies*, pp. 109–115, 2012.
- [16] K. Selvakumar, R. S. Vignesh, and R. Vijayabalan, “Shuffled Frog Leaping Algorithm for solving profit based unit commitment problem,” in *Proceedings of the International Conference on Computing, Communication and Applications (ICCCA '12)*, pp. 1–6, February 2012.
- [17] W. Z. Liu, S. Li, Y. Xiang, and X. Y. Tang, “Virtual network embedding based on shuffled frog leaping algorithm in TUNIE,” *International Journal of Advancements in Computing Technology*, vol. 3, no. 11, pp. 402–409, 2011.
- [18] X. D. Zhang, Y. Q. Bao, J. Xi, L. Zhao, and C. R. Zou, “Whispered speech emotion recognition based on improved shuffled frog leaping algorithm neural network,” *Journal of Convergence Information Technology*, vol. 7, no. 19, pp. 121–130, 2012.
- [19] X. Zhang, Y. Zhang, Y. Shi, L. Zhao, and C. Zou, “Power control algorithm in cognitive radio system based on modified Shuffled Frog Leaping Algorithm,” *AEU—International Journal of Electronics and Communications*, vol. 66, no. 6, pp. 448–454, 2012.
- [20] C. Chen, R. Xie, and X. N. He, “Research of multi-stage charging method for on-board lithium battery of electric vehicles,” *Journal of Mechanical & Electrical Engineering*, vol. 28, no. 7, pp. 887–890, 2011 (Chinese).

## Research Article

# The Study of Reinforcement Learning for Traffic Self-Adaptive Control under Multiagent Markov Game Environment

Lun-Hui Xu,<sup>1,2</sup> Xin-Hai Xia,<sup>1,3</sup> and Qiang Luo<sup>1</sup>

<sup>1</sup> School of Civil Engineering and Transportation, South China University of Technology, Guangzhou 510640, China

<sup>2</sup> School of Electronic Engineering and Automation, Jiangxi University of Science and Technology, Ganzhou, Jiangxi 341000, China

<sup>3</sup> School of Port and Shipping Management, Guangzhou Marine Institute, Guangzhou 510725, China

Correspondence should be addressed to Xin-Hai Xia; [xiainhai@126.com](mailto:xiainhai@126.com)

Received 25 February 2013; Revised 12 August 2013; Accepted 26 August 2013

Academic Editor: Orwa Jaber Housheya

Copyright © 2013 Lun-Hui Xu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Urban traffic self-adaptive control problem is dynamic and uncertain, so the states of traffic environment are hard to be observed. Efficient agent which controls a single intersection can be discovered automatically via multiagent reinforcement learning. However, in the majority of the previous works on this approach, each agent needed perfect observed information when interacting with the environment and learned individually with less efficient coordination. This study casts traffic self-adaptive control as a multiagent Markov game problem. The design employs traffic signal control agent (TSCA) for each signalized intersection that coordinates with neighboring TSCAs. A mathematical model for TSCAs' interaction is built based on nonzero-sum Markov game which has been applied to let TSCAs learn how to cooperate. A multiagent Markov game reinforcement learning approach is constructed on the basis of single-agent Q-learning. This method lets each TSCA learn to update its Q-values under the joint actions and imperfect information. The convergence of the proposed algorithm is analyzed theoretically. The simulation results show that the proposed method is convergent and effective in realistic traffic self-adaptive control setting.

## 1. Introduction

As car ownership rates and traffic volume have steadily increased over the last decades, existing road infrastructure today is often strained nearly to its limits. Continuous expansion of this infrastructure, however, is not possible or even desirable due to spatial, economic, and environmental reasons. It is therefore of paramount importance to try to optimize the flow of traffic in a given infrastructure. Traffic self-adaptive control of multiple intersections is synergetic and has the potential to significantly alleviate traffic congestion in urban transportation networks as opposed to the commonly used fixed timing and actuated control systems. Existing traditional traffic adaptive control systems such as TRANSYT, SCOOT, SCATS, and sophisticated dynamic programming approach [1] do not have a mechanism for learning from feedback on the quality of their model, which may lead to systematic errors.

Several researchers have employed classical control methods such as fuzzy logic [2], neural networks [3], and evolutionary algorithms [4] to traffic self-adaptive control. These

methods perform well but cannot be adapted to the changing characteristics of traffic flow. Reinforcement learning (RL) [5] is able to perpetually learn and improve the service over time. Multiagent reinforcement learning is an extension of RL to multiple agents in stochastic environment. The decentralized traffic control problem is an excellent test for multiagent reinforcement learning due to the inherited dynamics and stochastic nature of the traffic system [6–9].

There are two shortages in the application of multiagent reinforcement learning to traffic self-adaptive control problem as discussed below.

- (1) *With Less Efficient Coordination.* The majority of the previous studies consider independent learning agents which do not include any explicit mechanism for coordination [6–14]. Only a few previous studies consider coordination mechanism between the learning agents. Kuyer et al. [15] consider explicit two-level coordination mechanism between the learning agents that extends Wiering [6] using the coordination graphs. Max-plus algorithm is used to estimate the

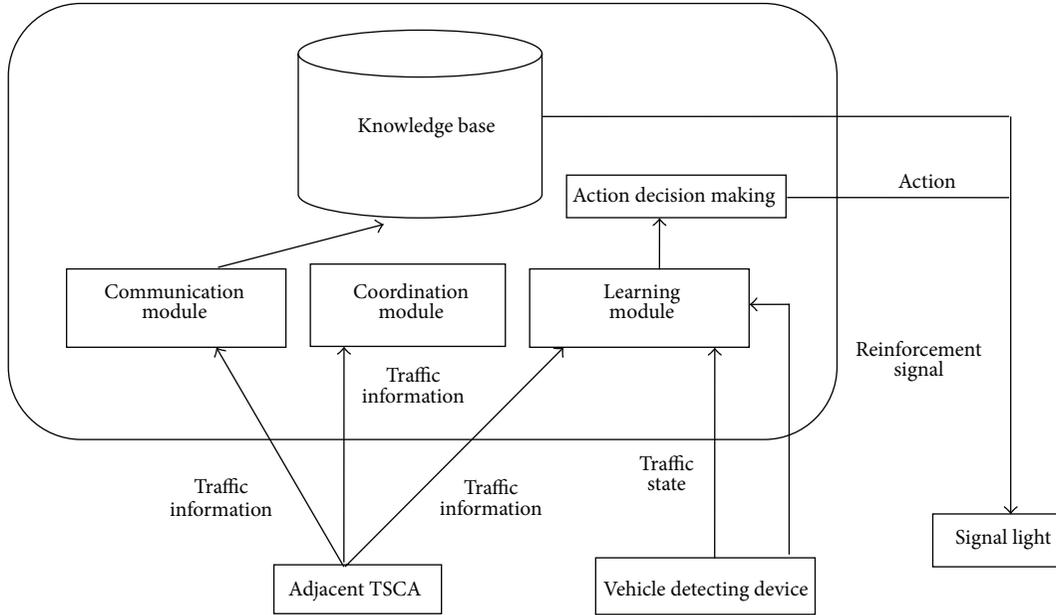


FIGURE 1: The structure model of TSCA.

optimal joint action by sending locally optimized messages among connected agents. However, Max-plus algorithm is computationally demanding, and therefore the agents report their current best action at any time even if the action found so far may be sub-optimal. Tantawy and Abdulhai [16] presented multiagent reinforcement learning for integrated network of adaptive traffic signal controllers (MARLIN-ATSC) which maintains a coordination mechanism (indirect coordination and direct coordination) between agents without compromising the dimensionality of the problem. Indirect coordination is realized by best-response multiagent learning in nonstationary environments, and the direct coordination is typically based on communication.

- (2) *On the Assumption of the Complete Knowledge.* In addition, since the traffic environment is changing in time and intersection cannot fully understand other intersections' information such as traffic arrival rate, vehicle queue, or delays, it is over idealized that the utility matrix of each agent is public; that is, perfect observed information is required when agents are interacting with the environment. According to this assumption, agents may select individual actions that are locally optimal but that together result in global inefficiencies. So this assumption is not too realistic.

It is argued that the use of a model-based RL approach adds unnecessary complexities compared with using model-free Q-learning. To overcome the upper deficiencies of the previous approach, this paper conducts a multiagent Markov game reinforcement learning method for optimizing traffic self-adaptive control. We define a TSCA for each signalized intersection that coordinates with neighboring agents. A

mathematical model for TSCAs' interaction is built based on nonzero-sum Markov game which has been applied to let TSCAs learn how to cooperate. A multiagent Markov game reinforcement learning approach is constructed on the basis of single-agent Q-learning. This method let each TSCA learn to update its Q-values under the joint actions and imperfect information. Convergence and effectiveness of the improving algorithm are verified.

## 2. Structure Model for TSCA

We defined a TSCA for each urban signalized intersection that takes the charge of controlling all signal phases. Its main function is to establish corresponding control strategy which will be implemented by signal light according to the current traffic state of both of its own and of its neighbors [14]. Therefore, the intersection's traffic flow conditions are improved. Figure 1 shows the structure model for TSCA. The reinforcement signal is a reward function  $r^i$  which will be defined in Section 3.

The structure model of TSCA has been shown in Figure 1. As we can see in Figure 2, it is mainly composed of learning module, action decision making module, communication module, and coordination module. Learning module infers whether there are reasonable regulations from real-time observational data. If some reasonable regulations exist, learning module executes these regulations and determines signal control plan. Coordination module analyzes the present traffic state of TSCA to decide if it is necessary to send messages to the adjacent TSCAs, and it deals with the TSCA coordination. Communication module is mostly responsible for the communication with the adjacent TSCAs. Action decision making module resolves function of reasoning and decision making of TSCA.

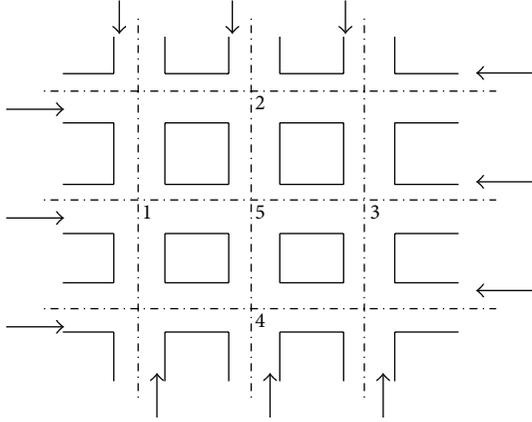


FIGURE 2: Traffic network used in the simulations.

In most cases, average delay time is sufficient to determine intersection's relative traffic performance; that is, lower average delay time implies lower ratio of stopped vehicles and total queue length. Vehicle's delay of coordinated controlled intersection is composed of normal delay, random delay, and oversaturation delay which is the same as isolated controlled ones. The computation of normal delay needs vehicle arriving and leaving graph. Since every incoming traffic flow of an intersection is up to green light time and released ratio of upstream intersections, arrival rate is not a constant but a time-varying functional expression. For random over saturation delay, the degree of traffic flow's random fluctuation within each cycle in coordinated control is far away less than that in isolated control, so delay value will decrease. Intersection's delay time can adopt the following transient functional model [17]:

$$\bar{d} = \begin{cases} \frac{C(1-\lambda)^2}{2(1-\gamma)} + \frac{N_0x}{q}, & x < 1, \\ \frac{R}{2} + \frac{N_0x}{q}, & x \geq 1, \end{cases} \quad (1)$$

$$N_0 = \begin{cases} \frac{q_c T}{4} \left[ (x-1) + \sqrt{(x-1)^2 + \frac{6(x-x_0)}{q_c T}} \right], & x > x_0, \\ 0, & x \leq x_0, \end{cases} \quad (2)$$

where  $\bar{d}$  is the average delay time per vehicle (s/pcu);  $C$  is a signal cycle length (s);  $\lambda$  is a split;  $\gamma$  is a ratio of flow volume;  $N_0$  is the average length of over saturation stopped fleet (pcu);  $q_c$  is the intersection's traffic capacity (pcu/h);  $T$  is a time interval (h);  $x$  is the intersection's saturation;  $q$  is the vehicle's arrival rate (pcu/h); and  $R$  is the length of red light time.

### 3. Mathematical Model for TSCAs' Interaction Based on Nonzero-Sum Markov Game

Game theory is the best mathematical tool to study human society's interaction. The interaction between TSCAs comes down to game model [18]. In the dynamic traffic signal

control system of multiple intersections, each intersection's signal timing scheme not only affects directly its neighbor's traffic but also indirectly affects nonneighboring intersection's traffic. Although, in traffic networks, the TSCA is incapable of observing the conditions of the entire network, it is possible to observe the conditions of the neighboring TSCAs. In order to avoid the excessive complexity and frequency of TSCAs' interaction brought by the increase in the number of controlled intersections, we limit that every TSCA to only interact with the adjacent one. Since two adjacent TSCAs can form a coalition to acquire a whole and local optimal performance in view of the openness of bilateral information, the interaction between two adjacent TSCAs conforms to a two-matrix nonzero-sum cooperative game [19].

Markov decision process (MDP) is widely researched which represents the problem about single agent in multiple states. By contrast, two matrix games were used to solve the problem for multiple agents in single state. Markov game can be regarded as the combination of MDP and two-matrix game which defines the frame of multiple agents and multiple environment states [18–20]. Since the traffic environments which TSCA is confronted with have the characteristics of dynamic, complexity, uncertainty, and openness,  $n$ -player nonzero-sum Markov game is suitable to be used to establish the interaction model for TSCAs.

An  $n$ -player nonzero-sum Markov game can be defined by a tuple  $M = \langle N, S, A, p, r \rangle$ , where the parameters  $N, S, A, p, r$  are explained in detail as follows according to traffic self-adaptive control.

$N$  is a set of finite interactive TSCAs.

$S = S^1 \times S^2 \times \dots \times S^i \times \dots \times S^N$  is a set of finite states. Let  $s^i$  be a local state of TSCA  $i$ ;  $s = (s^1, s^2, \dots, s^n)$  is a global state which is decomposed into local states.  $s^i$  is defined by a vector of two components. The first component is the position for the first vehicle approaching TSCA  $i$  from the directions of west, north, east, and south. Since the size of the state space grows rapidly if the accurate distance from the vehicle to TSCA represents local state, all lanes connected to TSCA are divided into a certain number of equal sections. The sections are sequentially encoded from the vehicle nearest to TSCA. So, the section's code can be used to define the local state  $s^i$ . The second component is the maximum queue lengths associated with each direction defined as follows [16]:

$$\max_{l \in L_i} q_l^k, \quad (3)$$

where  $q_l^k$  is the number of queued vehicles in lane  $l$  at time  $k$ . A vehicle is considered at a queue if its speed is below a certain speed threshold ( $Sp_{thr}$ ).  $q_l^k$  is computed as follows:

$$q_l^k = q_l^{k-1} + \sum_{v \in V_l^k} q_v^k, \quad (4)$$

$$q_v^k = \begin{cases} 1, & \text{if } Sp_v^{k-1} > Sp_{thr} \text{ and } Sp_v^k \leq Sp_{thr}, \\ -1, & \text{if } Sp_v^{k-1} \leq Sp_{thr} \text{ and } Sp_v^k > Sp_{thr}, \\ 0, & \text{if } Sp_v^{k-1} \leq Sp_{thr} \text{ and } Sp_v^k \leq Sp_{thr}, \end{cases}$$

where  $V_l^k$  is the set of vehicles traveling on lane  $l$  at time  $k$ .

$A = A^1 \times A^2 \times \dots \times A^i \times \dots \times A^N$  is a set of joint signal timing actions for multiple TSCAs, where  $A^i$  is the subset of finite signal timing actions for TSCA  $i$ .  $a^i$  represents the current phase duration which depends on intersection's phase and the adjustment of green time for the current green phase,  $a^i \in A^i$ . The adjustment of green time is to be determined by the relation of section's length and vehicle's velocity, for example, {green time plus 1 s, green light time plus 2 s, green time minus 1 s, green time minus 2 s, unchangeably}. Phase timing signal scheme generically consists of east-west straight and right turn, south-north straight and right turn, east-west left turn, and south-north left turn.  $a$  represents a joint signal timing action for the  $n$  TSCAs,  $a \in A$ .

$p : S \times A \rightarrow [0, 1]$  is a state transition function mapping a present state  $s$  and a joint signal timing action  $a$  to a probability over states.

$r = \{r^1, r^2, \dots, r^i, \dots, r^N\}$  is a set of reward functions for all TSCAs, where  $r^i : S \times A \rightarrow R$  is a reward function for TSCA  $i$  mapping state-action tuples to immediate scalar rewards.  $r^i(s, a)$  represents a reward function for TSCA  $i$  when the TSCAs take the joint action  $a$  in the state  $s$ .  $r^i(s, a)$  can be expressed by the division value between total volume for passing vehicles and accumulative waiting time.

Suppose that the traffic flow is random and corresponds to Poisson distribution. When green light of TSCA  $i$  in south-north direction is open,  $r^i(s, a)$  is described in a mathematical equation as follows:

$$r^i(s, a) = \frac{\delta_n(a^i - t_n) + \delta_s(a^i - t_s)}{(1/2)k\delta_w(a^i - t_w)^2 - (1/2)k\delta_e(a^i - t_e)^2}. \quad (5)$$

When green light of TSCA  $i$  in east-west direction is open,  $r^i(s, a)$  is computed as follows:

$$r^i(s, a) = \frac{\delta_w(a^i - t_w) + \delta_e(a^i - t_e)}{(1/2)k\delta_n(a^i - t_n)^2 - (1/2)k\delta_s(a^i - t_s)^2}, \quad (6)$$

where  $a^i$  is a signal timing action for TSCA  $i$  and  $t_w, t_n, t_e$ , and  $t_s$  represent the time to reach TSCA  $i$  for the first vehicle incoming but not passing TSCA  $i$  from the directions of west, north, east, and south, respectively, beginning with the alternation of red and green light. If  $a^i < t$  ( $t$  represents  $t_w, t_n, t_e$ , or  $t_s$ ), we set  $a^i - t = 0$ .  $\delta_w, \delta_n, \delta_e$ , and  $\delta_s$  represent the intensity for Poisson flow of vehicles incoming TSCA  $i$  from the directions of west, north, east, and south, respectively.  $\delta$  depends on the state and timing signal action of TSCAs adjacent to TSCA  $i$ . It can generalize from the average value of statistical historical data.  $k$  is a positive constant which represents the degree of punishment.

Let  $\pi^i$  be the probability distributions over action set  $A^i$  of TSCA  $i$ . With each  $s \in S$ , there is an  $n$ -player game  $\Gamma = \{(\pi^1, \pi^2, \dots, \pi^n)\}$ .  $V^i(s, \pi^1, \pi^2, \dots, \pi^n)$  is TSCA  $i$ 's total discounted reward in state  $s$  and under joint strategy  $(\pi^1, \pi^2, \dots, \pi^n)$ . Suppose that  $\prod = \{\pi^i, i \in N\}$  is a joint strategy when each TSCA selects action  $a^i$  with a probability  $\pi^i$ . We define  $\prod^{-i} = \{\pi^j, j \neq i, i, j \in N\}$ ; for each given

$\prod^{-i}$ , TSCA  $i$  chooses a corresponding optimal strategy  $\pi^{*i} = \operatorname{argmax} V\{\prod^{-i} \cup \pi^i\}$ .

In an  $n$ -player nonzero-sum Markov game under mixed strategy, a Nash equilibrium point is a tuple of  $n$  strategies  $(\pi^{*1}, \dots, \pi^{*i}, \dots, \pi^{*n})$  such that, for all  $s \in S$  and  $i = 1, \dots, n$ ,

$$V^i(\pi^{*1}, \dots, \pi^i, \dots, \pi^{*n}) \leq V^i(\pi^{*1}, \dots, \pi^{*i}, \dots, \pi^{*n}), \quad (7)$$

$$\forall \pi^i \in \Delta(A^i),$$

where  $\Delta(A^i)$  is the space of probability distributions over the TSCA  $i$ 's actions.

In the nonzero-sum Markov game, TSCA  $i$  can gain more rewards via cooperation than that under independent action. A Nash equilibrium point can be reached when none TSCAs gain more optimal policy.

## 4. Multiagent Markov Game Reinforcement for Traffic Self-Adaptive Control

4.1. *Single-Agent Q-Learning.* Q-learning which was presented by Watkins defines a learning method within a Markov decision process [21]. The basic idea of Q-learning is that we can define a function  $Q$  such that

$$Q^*(s, a) = r(s, a) + \gamma \sum_{s'} p(s' | s, a) V(s', \pi^*), \quad (8)$$

$$V(s, \pi^*) = \max_a Q^*(s, a). \quad (9)$$

By this definition,  $\gamma \in [0, 1]$  is a discount factor and is used to discount future rewards.  $p(s' | s, a)$  is the probability of transiting to state  $s'$  after taking action  $a$  in state  $s$ . A solution  $\pi^*$  that satisfies (9) is guaranteed to be an optimal policy.  $Q^*(s, a)$  is the total discounted reward of taking action  $a$  in state  $s$  and then following the optimal policy thereafter.

If  $Q^*(s, a)$  is given, then the optimal policy  $\pi^*$  can be found by simply identifying the action that maximizes  $Q^*(s, a)$  under the state  $s$ . The problem is then reduced to finding the function  $Q^*(s, a)$  instead of searching for the optimal value of  $V(s, \pi^*)$ .

Q-learning provides us with a simple updating procedure, in which the TSCA starts with arbitrary initial values of  $Q(s, a)$  for all  $s \in S, a \in A$  and updates the Q-values as follows:

$$Q_{t+1}(s, a) = (1 - \alpha) Q_t(s, a) + \alpha \left[ r_t + \gamma \max_b Q_t(s', b) \right], \quad (10)$$

where  $\alpha \in [0, 1]$  is the learning rate sequence.

Watkins and Dayan proved that sequence (8) converges to  $Q^*(s, a)$  under the assumption that all states and actions have been visited infinitely often and the learning rate satisfies certain constraints [21].

Even in single-agent Q-learning approach that is proven to optimally converge to the joint policy, each TSCA has to keep a set of tables whose size is exponential in the number of agents:  $|A_1| \times \dots \times |A_n| \times |S_1| \times \dots \times |S_n|$ . In addition to the dimensionality issue, the method requires each TSCA to observe

the state of the whole system which is infeasible in the case of transportation networks. Single-agent  $Q$ -learning takes other TSCAs as a part of environment and it updates future rewards based on merely the TSCA's own maximum payoff regardless of other TSCAs' actions. In MAS, we adopted a multiagent Markov game reinforcement method in which each TSCA updates its  $Q(s, a)$  according to immediate reward by interacting with other TSCAs and observing actions taken by all other TSCAs and others' rewards.

**4.2. Multiagent Markov Game Reinforcement Learning for Traffic Self-Adaptive Control.** In this method, at each time  $t$ , TSCA  $k$  updates its own  $Q$ -values  $Q_t^k$  and learn about other adjacent TSCAs'  $Q$ -values  $Q_t^j$ , ( $j \in \{1, \dots, n\}$ ,  $j \neq k$ ), by observing its own reward, actions taken by neighboring TSCAs, neighbor's rewards, and the new state  $s'$ . In the state  $s_t$ , TSCA  $k$  calculates a Nash equilibrium  $\pi^1(s_t), \dots, \pi^k(s_t), \dots, \pi^n(s_t)$  for the stage game  $(Q_t^1(s_t), \dots, Q_t^k(s_t), \dots, Q_t^n(s_t))$ . Then, TSCA  $k$ 's value function  $V_t^k(s_t)$  is its payoff in state  $s_t$  for the selected equilibrium which can be computed as follows:

$$\begin{aligned} V_t^k(s_t) &= \text{Nash } Q_t^k(s_t) \\ &= \pi^1(s_t) \cdots \pi^k(s_t) \cdots \pi^n(s_t) Q_t^k(s_t). \end{aligned} \quad (11)$$

TSCA  $k$  updates its  $Q$ -values according to

$$\begin{aligned} &Q_{t+1}^k(s_t, a_t^1, \dots, a_t^k, \dots, a_t^n) \\ &= (1 - \alpha_t) Q_t^k(s_t, a_t^1, \dots, a_t^k, \dots, a_t^n) \\ &\quad + \alpha_t [r_t^k + \gamma \text{Nash } Q_t^k(s_{t+1})] \\ &= (1 - \alpha_t) Q_t^k(s_t, a_t^1, \dots, a_t^k, \dots, a_t^n) \\ &\quad + \alpha_t [r_t^k + \gamma \pi^1(s_{t+1}) \cdots \pi^k(s_{t+1}) \cdots \\ &\quad \quad \pi^n(s_{t+1}) Q_t^k(s_{t+1})] \\ &= (1 - \alpha_t) Q_t^k(s_t, a_t^1, \dots, a_t^k, \dots, a_t^n) \\ &\quad + \alpha_t \left[ r_t^k + \gamma \prod_i \pi^i(s_{t+1}) Q_t^k(s_{t+1}) \right]. \end{aligned} \quad (12)$$

Information about other TSCAs'  $Q$ -values is not given, so TSCA  $k$  must learn about them too. TSCA  $k$  updates the beliefs about TSCA  $j$ 's  $Q$ -function according to the same rule (12) which is applied to its own,

$$\begin{aligned} &Q_{t+1}^j(s_t, a_t^1, \dots, a_t^k, \dots, a_t^n) \\ &= (1 - \alpha_t) Q_t^j(s_t, a_t^1, \dots, a_t^k, \dots, a_t^n) \\ &\quad + \alpha_t [r_t^j + \gamma \text{Nash } Q_t^j(s_{t+1})] \\ &= (1 - \alpha_t) Q_t^j(s_t, a_t^1, \dots, a_t^k, \dots, a_t^n) \end{aligned}$$

$$\begin{aligned} &+ \alpha_t [r_t^j + \gamma \pi^1(s_{t+1}) \cdots \pi^k(s_{t+1}) \cdots \\ &\quad \pi^n(s_{t+1}) Q_t^j(s_{t+1})] \\ &= (1 - \alpha_t) Q_t^j(s_t, a_t^1, \dots, a_t^k, \dots, a_t^n) \\ &\quad + \alpha_t \left[ r_t^j + \gamma \prod_i \pi^i(s_{t+1}) Q_t^j(s_{t+1}) \right]. \end{aligned} \quad (13)$$

Note that  $(\pi^1, \dots, \pi^k, \dots, \pi^n)$  is a joint mixed strategy to a Nash equilibrium point.  $r_t^k$  and  $s_{t+1}$  are TSCA  $k$ 's observed information. In order to update  $Q$ -values, TSCA  $k$  would know the priori knowledge about other TSCAs' strategy value  $\pi^i$  ( $i \neq k$ ). That is to say, we should solve a set composed of (13) when  $i = 1, \dots, k-1, k+1, \dots, n$ . This is an  $n$ -order nonlinear problem which has no practical solutions. In addition, since the TSCAs' observed information is imperfect, and in order to avoid the curse of dimensionality, we use probability statistics and Bayes method to estimate beliefs about other TSCAs' mixed strategies. And in such coordination mechanism, TSCA can reach a unique equilibrium.

Let us define that TSCA  $j$  conjecture TSCA  $i$  takes the action  $a^i$  in the probability of  $p^k(a^i)$ . The probability  $p^k(a^i)$  can be calculated as follows via Boltzmann formula:

$$p^k(a^i) = \frac{e^{Q(s_t, a^i)/T}}{\sum_{a \in A^i} e^{Q(s_t, a)/T}}, \quad (14)$$

where  $T$  is a temperature parameter which reflects exploring degree and decreases with time.

Each TSCA takes its own actions in state  $s_t$ ; then TSCA  $k$  observes other TSCAs' taken actions and new state  $s_{t+1}$ ; after that, TSCA  $k$  updates the belief about other TSCAs' actions. According to Bayes formula, the belief about TSCA  $j$ 's actions can be computed as follows:

$$p(a^i | a^k, s_{t+1}) = \frac{p(s_{t+1} | a^i, a^k) p(a^i)}{p(s_{t+1} | a^k)}, \quad (15)$$

where  $p(s_{t+1} | a^i, a^k)$  is the probability of transiting to state  $s_{t+1}$  after TSCA  $i$  and TSCA  $k$  take joint actions in state  $s_t$  and  $p(s_{t+1} | a^k)$  is the probability of transiting to state  $s_{t+1}$  after TSCA  $k$  takes its own action independently in state  $s_t$ . The probability of  $p(s_{t+1} | a^k)$  and  $p(s_{t+1} | a^i, a^k)$  can be acquired from environment knowledge.  $p(a^i)$  represents the probability in which TSCA  $i$  will take the action  $a^i$ .  $p^k(a^i)$  can similarly replace the estimate of  $p^k(a^i)$ .

Furthermore, TSCA  $k$ 's belief about other multiple TSCAs' joint actions can be generalized from (15) as follows:

$$p(a^1, \dots, a^n | a^k, s_{t+1}) = \prod_{i \neq k} p(a^i | a^k, s_{t+1}). \quad (16)$$

According to the analysis above, the multiagent Markov game reinforcement algorithm is summarized as follows by taking TSCA  $k$  for example.

Initialize

- (1) Let  $t = 0$ ; get the initial state  $s_0$ . For any  $s \in S$  and any  $a^k \in A^k$ , let  $\pi^k(s, a^k) = 1/|A|$  and  $Q_t^k(s, a_t^1, \dots, a_t^k, \dots, a_t^n) = 0$ .

Loop

- (2) Choose action  $a_t^k$ : TSCA  $k$  chooses the action  $a_t^k$  by the probability of  $\pi^k(s, a^k)$  which is the best-response mixed policy of the matrix game  $Q_t^k(s, a^1, \dots, a^n)$  under the joint actions of other TSCAs.
- (3) Observe  $r_t^1, \dots, r_t^n; a_t^1, \dots, a_t^n$ ; and  $s_{t+1} = s'$ .
- (4) TSCA  $k$  updates the belief about other TSCAs' actions according to Bayes formula (15).
- (5) Update  $Q_{t+1}^k$  for  $k = 1, \dots, n$ . Consider

$$\begin{aligned} Q_{t+1}^k(s_t, a_t^1, \dots, a_t^k, \dots, a_t^n) \\ = (1 - \alpha_t) Q_t^k(s_t, a_t^1, \dots, a_t^k, \dots, a_t^n) \\ + \alpha_t [r_t^k + \gamma p(a^1, \dots, a^n | a^k, s_{t+1}) Q_t^k(s_{t+1})], \end{aligned} \quad (17)$$

where  $p(a^1, \dots, a^n | a^k, s_{t+1})$  is defined in (16).

- (6) Let  $t = t + 1$  return to (2).

## 5. Convergence Proof

*Assumption 1.* For any multiagent game  $(Q^1(s), \dots, Q^N(s))$ , its Nash equilibrium point  $(\pi^1, \pi^2, \dots, \pi^N)$  has the following properties.

- (i) If the Nash equilibrium does not reach a global optimum, then the TSCA which takes the policy of Nash equilibrium will get more payoffs when the other TSCAs' policies deviate from Nash equilibrium:

$$\begin{aligned} \prod_{j \in N} \pi^j(s) \cdot \sum_{i \in N} Q^i(s) \leq \sum_{i \in N} Q^i(s) \cdot \pi^i(s) \cdot \prod_{j \in N, j \neq i} \hat{\pi}^j(s), \\ \forall \hat{\pi}^j(s) \in \Delta(A^j). \end{aligned} \quad (18)$$

- (ii) If the Nash equilibrium reaches a global optimum, then

$$\begin{aligned} \prod_{j \in N} \pi^j(s) \cdot \sum_{i \in N} Q^i(s) \geq \sum_{i \in N} Q^i(s) \cdot \prod_{j \in N} \hat{\pi}^j(s), \\ \forall \hat{\pi}^j(s) \in \Delta(A^j). \end{aligned} \quad (19)$$

*Assumption 2.* Learning sequence  $\{\alpha_t\}$  satisfies the following:

- (i)  $0 \leq \alpha_t \leq 1$ ,  $\sum_{t=1}^{\infty} \alpha_t = \infty$ ,  $\sum_{t=1}^{\infty} \alpha_t^2 < C < \infty$ , and the latter two hold uniformly and with probability 1, where  $C$  is constant,

- (ii) if  $(s, a^i, a^{-i}) \neq (s_t, a_t^i, a_t^{-i})$ , then  $\alpha_t(s, a^i, a^{-i}) = 0$ , where  $i \in N$ . Under Assumption (i), given a finite MDP, the Q-learning algorithm proposed by [21] converges with probability 1 to the optimal Q-function. The second item in Assumption 2 states that the agent updates only the Q-function element corresponding to current state  $s_t$  and actions  $a_t^1, \dots, a_t^n$ .

**Lemma 3** (see [22]). *Under Assumption 2, iterative process  $Q_{t+1} = (1 - \alpha_t)Q_t + \alpha_t w_t$  will converge to  $E(w_t | h_t, \alpha_t)$ , with probability 1, where  $h_t$  is the historical states and policies at the time  $t$ .*

**Lemma 4** (see [23]).  $\forall s \in S$ , assume that  $(\pi^1(s), \pi^2(s), \dots, \pi^N(s))$  is the Nash equilibrium point for the stage game  $(Q^1(s), Q^2(s), \dots, Q^N(s))$ , its Nash equilibrium payoff is  $(v^1(s, \pi^1, \pi^{-1}), v^1(s, \pi^2, \pi^{-2}), \dots, v^1(s, \pi^N, \pi^{-N}))$ , then the TSCA  $i$ 's Nash equilibrium payoff under joint actions is

$$\begin{aligned} Q^i(s, a^1, \dots, a^N) = r(s, a^1, \dots, a^N) \\ + \gamma \sum_{s'=1}^N p(s' | s, a^1, \dots, a^N) \\ \times v^i(s', \pi^1, \pi^2, \dots, \pi^N). \end{aligned} \quad (20)$$

**Lemma 5** (see [23]). *Given that  $P_t^i Q^i(s) = r^i + \gamma Q^i(s) \prod_{j \in N} \pi^j(s)$  where,  $\forall s \in S$ ,  $(\pi^1(s), \pi^2(s), \dots, \pi^N(s))$  is the mixed Nash equilibrium policy for the stage game  $(Q^1(s), Q^2(s), \dots, Q^N(s))$ , then  $P_t = (P_t^1, \dots, P_t^N)$  is a contract mapping.*

**Lemma 6** (see [22]). *Under Assumption 2, if the iterative process  $U_{t+1}(x) = (1 - \alpha_t(x))U_t(x) + \alpha_t(x)[P_t v^*](x)$  holds and  $P_t$  satisfies  $\|P_t V - P_t v^*\| \leq \gamma \|V - v^*\| + \lambda_t, \forall V$ , where  $0 \leq \gamma \leq 1$ ,  $\lambda_t \geq 0$  and converges to 0 with probability 1, then the following iterative process will converge to  $v^*$  with probability 1:*

$$V_{t+1}(x) = (1 - \alpha_t(x)) V_t(x) + \alpha_t(x) [P_t v_t](x). \quad (21)$$

Under the above assumptions and lemmas, if the Nash equilibrium point for the stage game is obtained, then proposed method converges to game Q-values of equilibrium point with probability 1.

**Theorem 7.** *Assume that the game sequence  $(Q_t^1(s), \dots, Q_t^N(s))$  for each  $s \in S(Q_t^1(s), \dots, Q_t^N(s))$  satisfies*

$$\begin{aligned} Q_{t+1}^i(s, a^1, \dots, a^N) = (1 - \alpha_t) Q_t^i(s, a^1, \dots, a^N) \\ + \alpha_t \left[ r_t^i + \gamma Q_t^i(s') \prod_{j \in N} \pi^j(s') \right], \end{aligned} \quad (22)$$

then the game sequence converges to  $Q$ -value of equilibrium point  $(Q^{1*}, \dots, Q^{N*})$  with probability 1, where the  $Q$ -value meets

$$Q^{i*}(s, a^1, \dots, a^N) = r^i(s, a^1, \dots, a^N) + \gamma \sum_{s'=1}^N p(s' | s, a^1, \dots, a^N) \times v^i(s', \pi^{1*}, \pi^{2*}, \dots, \pi^{N*}), \quad (23)$$

where  $i \in N$ ;  $(\pi^1(s'), \pi^2(s'), \dots, \pi^N(s'))$  is the mixed Nash equilibrium policy for the stage game  $(Q_t^1(s'), \dots, Q_t^N(s'))$  at the state  $s'$ ;  $(\pi^{1*}, \pi^{2*}, \dots, \pi^{N*})$  is the Nash equilibrium point for the game.

*Proof.* Under Lemma 4,

$$\|P_t^i Q^i - P_t^i Q^{i*}\| \leq \gamma \|Q^i - Q^{i*}\|. \quad (24)$$

Under Lemma 3,

$$Q_{i+1}^k(s, a^1, \dots, a^N) = (1 - a_t) Q_t^k(s, a^1, \dots, a^N) + a_t \left[ r_t^i + \gamma Q_t^i(s') \prod_{j \in N} \pi^j(s') \right] \quad (25)$$

converges to

$$E \left( r_t^i + \gamma Q^i(s) \prod_{j \in N} \pi^j(s) \right) = \sum_{s'} p(s' | s, a^1, \dots, a^N) \cdot \left( r_t^i(s, a^1, \dots, a^N) + \gamma Q^i(s') \prod_{j \in N} \pi^j(s') \right). \quad (26)$$

Defining mapping  $T^i$  as

$$(T^i Q^i)(s' | s, a^1, \dots, a^N) = \sum_{s'} p(s' | s, a^1, \dots, a^N) \cdot \left( r_t^i(s, a^1, \dots, a^N) + \gamma Q^i(s') \prod_{j \in N} \pi^j(s') \right), \quad (27)$$

then  $T^i Q^i = \sum_{s'} p(s' | s, a^1, \dots, a^N) \cdot P_t^i Q^i(s)$ . Under Lemma 5, since  $P_t^i$  is the contract mapping of  $Q^i$  and  $p(s' | s, a^1, \dots, a^N) \geq 0$ ,  $T^i$  is also the contract mapping of  $Q^i$  and  $Q^{i*}$  is the fixed point of mapping  $T^i$ . Consider

$$(T^i Q^{i*})(s, a^1, \dots, a^N) = \sum_{s'} p(s' | s, a^1, \dots, a^N)$$

$$\cdot \left( r_t^i(s, a^1, \dots, a^N) + \gamma Q^{i*}(s') \prod_{j \in N} \pi^{j*}(s') \right) = r_t^i(s, a^1, \dots, a^N) + \gamma \sum_{s'} p(s' | s, a^1, \dots, a^N) \cdot Q^{i*}(s') \prod_{j \in N} \pi^{j*}(s'). \quad (28)$$

Under Lemma 4,  $Q^{i*}(s') \prod_{j \in N} \pi^{j*}(s') = v^i(s', \pi^{1*}, \dots, \pi^{N*})$ , so  $Q^{i*} = T^i Q^i$ . Therefore the iterative formula

$$Q_{t+1}^k(s, a^1, \dots, a^N) = (1 - \alpha) Q_t^k(s, a^1, \dots, a^N) + \alpha \left[ r_t^i + \gamma Q^{i*}(s') \prod_{j \in N} \pi^{j*}(s') \right] \quad (29)$$

converges to  $T^i Q^i = Q^{i*}$  with probability 1. Under Lemma 6, formula (20) converges to  $Q^{i*}$ .  $\square$

## 6. Simulation

**6.1. Analysis of the Method's Convergence.** We consider the traffic network shown in Figure 2 used for the scenery to test the proposed approach described in Section 4. Paramics, a microscopic traffic simulator, is used to build the testbed network. The multiagent Markov game reinforcement learning algorithm is written in Matlab as a stand-alone application. The interaction between the multiagent Markov game reinforcement learning algorithm and the Paramics environment is implemented through the application programming interface (API) functions in Paramics.

$\delta$ , defined as the intensity of Poisson flow observed entering the traffic network, obeys uniform distribution in the interval of [5, 18]. The traffic network includes 5 intersections in which each direction has four lanes. Each intersection sets two phases. The length of the lanes is 60 m. The average velocity of the traffic flow is 2.5 m/s. Let  $\alpha = 0.1$ ,  $\gamma = 0.98$ ,  $k = 0.25$ , and  $T = 40$ . Since some general constraints posed by safety rules should be respected when designing the signal plan, we let minimum and maximum green time for each phase be 20 s and 90 s, respectively. The subsets of actions for TSCA 1, TSCA 2, TSCA 3, TSCA 4, and TSCA 5 are given by  $A^1 = \{15, 30, 40\}$ ,  $A^2 = \{30, 50\}$ ,  $A^3 = \{35, 45\}$ ,  $A^4 = \{25, 50\}$ , and  $A^5 = \{30, 35, 40, 55\}$ , respectively. Each lane is equally divided into a certain number of segments in an interval of 20 m. As a result, the values of joint state and actions are set as  $s = \langle \langle 1, 2, 3, 1 \rangle, \langle 2, 3, 1, 2 \rangle, \langle 2, 1, 2, 1 \rangle, \langle 1, 1, 3, 2 \rangle, \langle 1, 2, 3, 2 \rangle \rangle$ , and  $a = \langle 30, 30, 45, 25, 35 \rangle$ , respectively. Under the state  $s$  and in the taken joint action  $a$  given above, how the  $Q$ -values of TSCA 1, TSCA 2, and TSCA 5 vary with learning time, respectively, was shown in Figure 3.

As can be seen from Figure 3, the multiagent Markov game reinforcement learning approach presented in this paper is convergent; that is, it can reach a Nash equilibrium. In general conditions, an urban traffic network has a relatively stable flow of vehicles, so the time to solve such problem is acceptable.

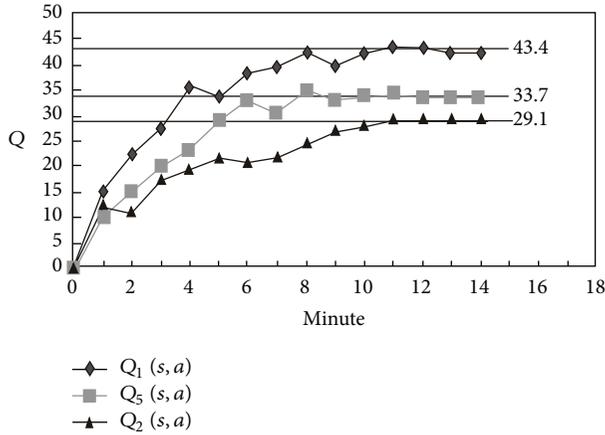


FIGURE 3: Q-values of TSCA 1, TSCA 2, and TSCA 5 per minute under  $s$  and in  $a$ .

In the traffic network shown in Figure 2, supposed sometime TSCA 5 is about to choose a red light timing action in north-south direction; then we specify that the local state of TSCA 1, 2, 3, 4 is  $s^1 = s^2 = s^3 = s^4 = \langle 1, 1, 1, 1 \rangle$ , respectively, so the finally learned Q-values of TSCA 5 are shown in Table 1.

**6.2. Analysis of the Method's Effectiveness.** We use the average delay time per vehicle ( $\bar{d}$ ) which was defined in (1) of Section 2 as the performance index of each method.

**6.2.1. In Comparison with LQF.** Local traffic consists of vehicles that cross a single intersection and then exit the network, thereby interacting with just one learning agent. According to [24], when the saturation is greater than 0.90, the level of intersection's service is unbearable. So we think that a highly saturated condition in this paper refers to one in which the saturation greatly exceeds 0.9, and as a result the road is congested and the service level is relatively poor. In this section, we compare the novel approach described in Section 4.2 to LQF traffic signal scheduling algorithm proposed in [25] for an isolated intersection. The LQF algorithm was designed for a signal control problem employing concepts drawn from the field of packet switching in computer networks. It utilized a maximal weight matching algorithm to minimize the queue sizes at each approach yielding significantly lower average vehicle delay through the intersection. The primary limitation of LQF is that every agent only considers its own local traffic volume and thus controls its traffic signals in isolation. Consequently, agents may select individual actions that are locally optimal but that together result in global inefficiencies. Therefore, we focus our experiments on comparisons between the novel method and LQF in highly saturated conditions.

In particular, we also consider the scenery shown in Figure 2. In such traffic network all routes contain at least two intersections, and destinations are selected uniformly, thereby eliminating local traffic. The scenery is challenging

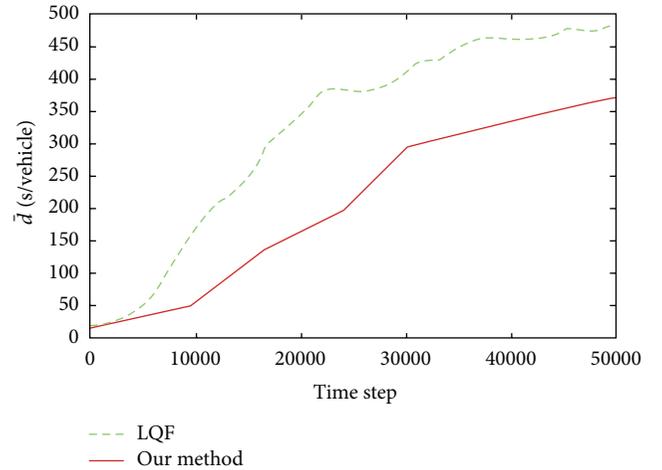


FIGURE 4: The results comparison with LQF.

TABLE 1: The learned Q-values of TSCA 5 in specified state.

Local state of TSCA 5	1, 1, 2, 1	1, 1, 2, 2	1, 1, 2, 3	1, 1, 3, 1
Max Q	231.4	278.9	297.4	211.8
Timing	60	30	25	25

and realistic, as it requires the methods to cope with an abundance of nonlocal traffic. The experiment is designed to test the hypothesis that, under highly saturated conditions, coordinated learning is beneficial when the amount of local traffic is small. If this hypothesis is correct, coordinated learning with multiagent Markov game reinforcement learning should substantially outperform LQF when most vehicles pass through multiple intersections.

We use the average delay time per vehicle ( $\bar{d}$ ) which is defined in (1) of Section 2 as the performance index of each method.

The result is averaged over 10 independent runs. Figure 4 shows the results from the nonuniform destinations and non-local traffic scenery. As can be seen from the figure, multiagent Markov game reinforcement learning substantially outperforms the other noncoordinated method. This result is not surprising since the lack of uniform destinations and local traffic create clear incentive for the TSCAs to learn to coordinate their actions. This approach allows the TSCAs to learn different state transition probabilities and value functions when the outbound lanes are congested. For example, the lane from intersection 1 to 5 is likely to become saturated as all traffic from edge nodes connected to intersection 1 must travel through it. When such saturation occurs, it is important for the two TSCAs to learn to coordinate since allowing incoming traffic to cross intersection 1 is pointless unless intersection 5 allows that same traffic to cross in a "green wave". The cost of including such congestion information is a larger state space and potentially slower learning. So, the simulation results show that multiagent Markov game reinforcement learning is effective.

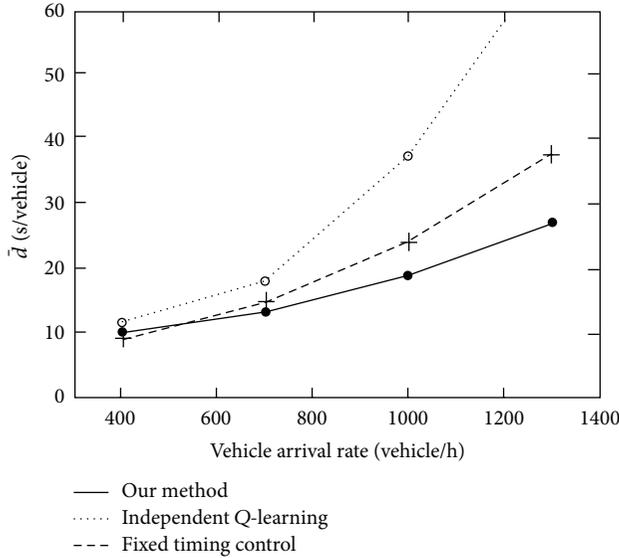


FIGURE 5: The results comparison with fixed timing control, independent Q reinforcement learning.

**6.2.2. In Comparison with Fixed Timing Control Approach and Independent Q Reinforcement Learning Approach.** In this section, we compare the novel approach described in Section 4.2 to fixed timing control and independent reinforcement learning. We consider the scenery shown in Figure 2 again. Assume that the vehicle arrival rate in the direction of north-south is

$$p_{sn} = 0.3 + 0.1 \cos\left(\frac{\pi t}{540}\right) \quad (30)$$

and the vehicle arrival rate in the direction of east-west is

$$p_{ew} = 0.1 + 0.5 \sin\left(\frac{\pi t}{540}\right). \quad (31)$$

The yellow light time length is 4 s; TSCA schedules traffic signal every two seconds. The learning time length is 5400 s. During the independent reinforcement learning, TSCA does not consider other TSCA's actions and states, the reinforcement signals from the belief allocation modules are only associated with their own states and actions, then the goal is to maximize the local rewards. In the fixed timing control, the signal cycle length is 120 s. The effective green time length in the direction of north-south and east-west is 56 s. Figure 5 shows the results.

The results show that in low traffic flow the differences between the effectiveness of the three control methods are not particularly evident. When the traffic flow increases gradually, the differences between the performance of the three methods are more and more apparent. Since independent Q-learning process does not take other TSCA's states and actions into consideration, it is not easy to achieve global optimum, which may lower the performance of the system. When the vehicle arrival rate is more than 1000 vehicle/h, independent Q-learning leads to a lot of heavy traffic. On the contrary, in our method, each TSCA must consider the influence of other

TSCA's states and actions, so the results have a certain amount of global properties.

Next, we analyze why multiagent Markov game reinforcement learning approach outperforms the other two approaches. Fixed timing control method can not be adapted to the changes of the traffic environment. In the independent Q-learning algorithm, each TSCA learns and decides at the local level (i.e., using its local state and local action) by using (10). Multiagent Markov game reinforcement learning method biases action selection toward actions that are likely to result in good rewards. The likelihood of good values is evaluated using models of the other agents estimated by the learner through observing their behavior in the past. The efficiency of multiagent Markov game reinforcement learning approach is more profound in cases of traffic fluctuations which assure the adaptability of the approach as the highly saturated condition triggers the TSCAs to coordinate their actions.

## 7. Conclusion

Previous work about urban traffic control used multiagent reinforcement learning, but the TSCAs selected only locally optimal actions without coordinating their behavior and needed perfect observed information when interacting with environment. In this paper, a multiagent Markov game reinforcement learning approach based on  $n$ -player nonzero-sum Markov game is designed for optimizing urban traffic on the basis of the analysis of TSCA's structure model and single-agent Q-learning. Theoretical analysis and experimental result show that the proposed method is convergent and effective. Multiagent Markov game reinforcement learning substantially outperforms the other non-coordinated method like LQE, fixed timing control, and independent reinforcement learning especially under highly saturated conditions when the amount of local traffic is small. It will be demonstrated that the novel method offers the capability to provide distributed control as needed for scheduling multiple intersections.

In future work, it would be interesting to incorporate the effects of driver behavior and transit signal priority in our framework. Moreover, the basic five-intersection network considered here will be expanded to include larger traffic networks and more extensive collaboration among TSCAs.

## Acknowledgments

The work described in this paper was supported by the Natural Science Foundation of China (nos. 61263024 and 51268017). The authors would like to thank the members of the academic team guided by Professor Lun-hui Xu for their advice.

## References

- [1] C. Diakaki, M. Papageorgiou, and K. Aboudolas, "A multivariable regulator approach to traffic-responsive network-wide signal control," *Control Engineering Practice*, vol. 10, no. 2, pp. 183–195, 2002.

- [2] P. G. Balaji, D. Srinivasan, and C.-K. Tham, "Coordination in distributed multi-agent system using type-2 fuzzy decision systems," in *Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ '08)*, pp. 2291–2298, Piscataway, NJ, USA, June 2008.
- [3] D. Srinivasan, M. C. Choy, and R. L. Cheu, "Neural networks for real-time traffic signal control," *IEEE Transactions on Intelligent Transportation Systems*, vol. 7, no. 3, pp. 261–272, 2006.
- [4] J. J. Sánchez, M. Galán, and E. Rubio, "Genetic algorithms and cellular automata: a new architecture for traffic light cycles optimization," in *Proceedings of the Congress on Evolutionary Computation (CEC '04)*, pp. 1668–1674, June 2004.
- [5] R. Sutton and A. Barto, *Reinforcement Learning: An Introduction*, MIT Press, Boston, Mass, USA, 1998.
- [6] M. Wiering, "Multi-agent reinforcement learning for traffic light control," in *Proceedings of the 17th International Conference on Machine Learning and Applications*, pp. 1151–1158, 2000.
- [7] B. Abdulhai, R. Pringle, and G. J. Karakoulas, "Reinforcement learning for true adaptive traffic signal control," *Journal of Transportation Engineering*, vol. 129, no. 3, pp. 278–285, 2003.
- [8] M. D. Pendrith, "Distributed reinforcement learning for a traffic engineering application," in *Proceedings of the 4th International Conference on Autonomous Agents*, pp. 404–411, ACM Press, Rockville, Md, USA, June 2000.
- [9] B. Bakker, M. Steingrover, R. Schouten, E. Nijhuis, and L. Kester, "Cooperative multi-agent reinforcement learning of traffic lights," in *Proceedings of the Workshop on Cooperative Multi-Agent Learning*, pp. 24–36, 2005.
- [10] A. Salkham, R. Cunningham, A. Garg, and V. Cahill, "A collaborative reinforcement learning approach to urban traffic control optimization," in *Proceedings of the IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT '08)*, pp. 560–566, Sydney, Australia, December 2008.
- [11] I. Arel, C. Liu, T. Urbanik, and A. G. Kohls, "Reinforcement learning-based multi-agent system for network traffic signal control," *IET Intelligent Transport Systems*, vol. 4, no. 2, pp. 128–135, 2010.
- [12] J. C. Medina and R. F. Benekohal, "Q-learning and approximate dynamic programming for traffic control—a case study for an oversaturated network," in *Proceedings of the Boar Annual Meeting*, 2012.
- [13] M. Abdoos, N. Mozayani, and A. L. C. Bazzan, "Traffic light control in non-stationary environments based on multi agent Q-learning," in *Proceedings of the 14th IEEE International Intelligent Transportation Systems Conference (ITSC '11)*, pp. 1580–1585, Washington, DC, USA, October 2011.
- [14] P. G. Balaji, X. German, and D. Srinivasan, "Urban traffic signal control using reinforcement learning agents," *IET Intelligent Transport Systems*, vol. 4, no. 3, pp. 177–188, 2010.
- [15] L. Kuyer, S. Whiteson, B. Bakker, and N. Vlassis, "Multiagent reinforcement learning for urban traffic control using coordination graph," in *Proceedings of the 19th European Conference on Machine Learning*, 2008.
- [16] S. E. Tantawy and B. Abdulhai, "Multi-agent reinforcement learning for integrated network of adaptive traffic signal controllers (MARLIN-ATSC)," in *Proceedings of the 15th International IEEE Annual Conference on Intelligent Transportation Systems (ITSC '12)*, pp. 319–326, Anchorage, Alaska, USA, 2012.
- [17] P. Koonce, "Traffic signal timing manual," US Department of Transportation FHWA-HOP-08-024, Federal Highway Administration, 2008.
- [18] M. L. Littman, "Markov games as a framework for multiagent reinforcement learning," in *Proceedings of the 11th International Conference on Machine Learning*, pp. 157–153, Morgan Kaufmann, Boston, Mass, USA, 1994.
- [19] I. Alvarez, A. Poznyak, and A. Malo, "Urban traffic control problem a game theory approach," in *Proceedings of the 47th IEEE Conference on Decision and Control (CDC '08)*, pp. 2168–2172, Cancún, Mexico, December 2008.
- [20] J. Hu and M. P. Wellman, "Nash Q-learning for general-sum stochastic games," *Journal of Machine Learning Research (JMLR)*, vol. 4, no. 6, pp. 1039–1069, 2003.
- [21] C. J. C. H. Watkins and P. Dayan, "Q-learning," *Machine Learning*, vol. 8, no. 3-4, pp. 279–292, 1992.
- [22] C. Szepesvári and M. L. Littman, "A unified analysis of value-function-based reinforcement-learning algorithms," *Neural Computation*, vol. 11, no. 8, pp. 2017–2060, 1999.
- [23] J. Filar and K. Vrieze, *Competitive Markov Decision Processes*, Springer, New York, NY, USA, 1997.
- [24] W. Wang, *Urban Traffic Planning*, Southeast University Press, Nanjing, China, 1999.
- [25] R. Wunderlich, C. Liu, I. Elhanany, and T. Urbanik II, "A novel signal-scheduling algorithm with quality-of-service provisioning for an isolated intersection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 9, no. 3, pp. 536–547, 2008.

## Research Article

# Application of Adaptive Extended Kalman Smoothing on INS/WSN Integration System for Mobile Robot Indoors

Xiyuan Chen,<sup>1,2</sup> Yuan Xu,<sup>1,2</sup> and Qinghua Li<sup>1,2</sup>

<sup>1</sup> School of Instrument Science and Engineering, Southeast University, Nanjing 210096, China

<sup>2</sup> Key Laboratory of Micro-Inertial Instrument and Advanced Navigation Technology, Ministry of Education, Nanjing 210096, China

Correspondence should be addressed to Yuan Xu; xyayuan@126.com

Received 3 July 2013; Accepted 10 October 2013

Academic Editor: Orwa Jaber Housheya

Copyright © 2013 Xiyuan Chen et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The inertial navigation systems (INS)/wireless sensor network (WSN) integration system for mobile robot is proposed for navigation information indoors accurately and continuously. The Kalman filter (KF) is widely used for real-time applications with the aim of gaining optimal data fusion. In order to improve the accuracy of the navigation information, this work proposed an adaptive extended Kalman smoothing (AEKS) which utilizes inertial measuring units (IMUs) and ultrasonic positioning system. In this mode, the adaptive extended Kalman filter (AEKF) is used to improve the accuracy of forward Kalman filtering (FKF) and backward Kalman filtering (BKF), and then the AEKS and the average filter are used between two output timings for the online smoothing. Several real indoor tests are done to assess the performance of the proposed method. The results show that the proposed method can reduce the error compared with the INS-only, least squares (LS) solution, and AEKF.

## 1. Introduction

Automation indoor mobile robots have increasingly been used in a wide range of applications [1]. The ability to obtain their navigation information (such as position and velocity) has become one key issue. Although the global positioning system (GPS) is widely used for navigation applications, it is essential for outdoor navigation, and there is also a growing need for accurate navigation information indoors [2]. Consequently, this topic has received significant scientific research attention over the past few decades.

In order to achieve accurate navigation information indoors, a number of methods for localization with various sensors and precision were proposed [1–3]. For instance, in [4], an RFID-based position and orientation measurement system for mobile objects was proposed by Shirehjini et al.; in [5] Park proposed an indoor location system using ZigBee; in [6], Saad et al. proposed high-accuracy reference-free ultrasonic location estimation. All the above-mentioned attempts employ reference node (RN) with known location to complete the localization of blind node (BN). Its principle

is similar to global positioning systems (GPS), but the communication technology used by Beacon-based solutions is short-range radio, such as WiFi, UWB, RFID, ZigBee, and ultrasound. The shortcoming of the above-mentioned attempts is that the localization accuracy has to maintain a high density of RNs, which is not useful for large localization area.

To outdoor navigation, in order to achieve continuous navigation, inertial navigation systems (INS) have been used for the compensation to the GPS outage since it is capable of providing positioning information independently [7]. For example, a novel hybrid of least squares support vector machine (LS-SVM) and Kalman filter for GPS/INS integration was proposed by Xu et al. in [8]. To indoor navigation, Ruiz et al. employed inertial measuring units (IMUs)/radio frequency identification (RFID) integration navigation for pedestrian indoor navigation in [2]. However, INS solution is poor in long-term self-contained navigation since the accuracy deteriorates with time [9, 10].

In the integrated system, the integration filter should be carefully designed since it is the core of system. As one of the most popular information fusion algorithms, Kalman

filter (KF) is widely used in integrated system. However, although it is able to achieve the optimal estimation of states in multiinput and multioutput (MIMO) systems [11], KF is not suitable for nonlinear systems since the noises of system and measurement should be corrupted by white noise and the state estimation is approached with the minimization of the covariance of the estimation error. Then, the extended KF (EKF) is proposed to overcome this problem by using Taylor series expansion [10]. However, the EKF is difficult to track the accurate state during the target's fast movement since it employs a fixed priori estimates for the process and measurement noise covariances during the whole estimation process [12]. Thus, the AEKF is proposed to update the covariance of process noise and measurement noise in current. In order to obtain high accuracy of information fusion, smoothing algorithms have been widely used in integrated navigation systems [13]. Rauch-Tung-Strieble smoothing (RTSS) is widely used in navigation applications due to its robustness and effectiveness [14]. Liu et al. proposed two-filter smoothing (TFS) and applied it in INS/GPS integration for postprocessing applications in [15]. Meanwhile, the results proposed in [15] show that the TFS has the advantage to be applicable in cases of nonlinear dynamics that may occur in some land-vehicle navigation (LVN) applications. TFS is performed by combining the results of forward Kalman filtering (FKF) and backward Kalman filtering (BKF) by minimizing the smoother error covariance.

This work proposed the design and implementation of adaptive extended Kalman smoothing (AEKS) on INS/WSN integration system for mobile robot indoors. In this mode, the adaptive extended Kalman filter (AEKF) is employed to improve the forward filtering output accuracy, and the back filter is used to smooth the forward filtering output. In order to achieve online smoothing, the AEKS and the average filter are used between two output periods. The remainder of the paper is organized as follows: Section 2 gives the adaptive extended Kalman smoothing for integration system. The real indoor tests and performance are illustrated in Section 3. Finally, the conclusions are given.

## 2. Adaptive Extended Kalman Smoothing for Integration System

**2.1. Integration Model.** Figure 1 displays the configuration of the integrated system. In this mode, the integrated model which is proposed in [16] is employed for the integrated system in this work. The continuous-time process model of the system is illustrated as follows:

$$\begin{bmatrix} \delta P_{E,k} \\ \delta V_{E,k} \\ \delta Acc_{E,k} \\ \delta P_{N,k} \\ \delta V_{N,k} \\ \delta Acc_{N,k} \end{bmatrix} \underset{\mathbf{x}_k}{}$$

$$= \underbrace{\begin{bmatrix} 1 & T & \frac{T^2}{2} & 0 & 0 & 0 \\ 0 & 1 & T & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & T & \frac{T^2}{2} \\ 0 & 0 & 0 & 0 & 1 & T \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}}_{\mathbf{A}} \underbrace{\begin{bmatrix} \delta P_{E,k-1} \\ \delta V_{E,k-1} \\ \delta Acc_{E,k-1} \\ \delta P_{N,k-1} \\ \delta V_{N,k-1} \\ \delta Acc_{N,k-1} \end{bmatrix}}_{\mathbf{x}_{k-1}} + \mathbf{W}_k, \quad (1)$$

where  $(\delta P_{E,k}, \delta P_{N,k})$ ,  $(\delta V_{E,k}, \delta V_{N,k})$ , and  $(\delta Acc_{E,k}, \delta Acc_{N,k})$  are the errors of position, velocity and accelerometer measured by INS in east and north direction.  $T$  is sample time;  $\mathbf{W}_k$  is the Gaussian process noise.

The observation vectors of the filter are formed by differencing the WSN and INS velocities ( $V^{INS}, V^{WSN}$ ) and the distances between the robot and the  $i$ th RN ( $d_i^{INS}, d_i^{WSN}$ ). The measurement equation at time  $k$  is illustrated as follows:

$$\underbrace{\begin{bmatrix} \Delta V_{E,k} \\ \Delta V_{N,k} \\ \Delta d_{1,k}^2 \\ \Delta d_{2,k}^2 \\ \vdots \\ \Delta d_{m,k}^2 \end{bmatrix}}_{\mathbf{y}_k} = \underbrace{\begin{bmatrix} \delta V_{E,k} \\ \delta V_{N,k} \\ h_{d_1} (\delta P_{E,k}, \delta P_{N,k}) \\ h_{d_2} (\delta P_{E,k}, \delta P_{N,k}) \\ \vdots \\ h_{d_m} (\delta P_{E,k}, \delta P_{N,k}) \end{bmatrix}}_{\mathbf{h}(\mathbf{x}_k)} + \mathbf{v}_k. \quad (2)$$

Here, the difference between  $(d_i^{INS})^2$  and  $(d_i^{WSN})^2$  is denoted as  $\Delta d_i^2$ , and it is expressed as follows:

$$\begin{aligned} \Delta d_i^2 &= (d_i^{INS})^2 - (d_i^{WSN})^2 \\ &= 2(P_E^{INS} - x_i) \delta P_E + 2(P_N^{INS} - y_i) \delta P_N \\ &\quad - (\delta P_E^2 + \delta P_N^2), \quad i = 1, 2, \dots, m. \end{aligned} \quad (3)$$

And  $(\Delta V_E, \Delta V_N)$  is the difference of the WSN and INS velocities in east and north direction, respectively, and  $\mathbf{v}_k$  is the Gaussian process noise. It is assumed that  $\omega_k$  and  $\mathbf{v}_k$  are independent zero-mean white Gaussian sequences with covariances  $\mathbf{Q}$  and  $\mathbf{R}$ .

**2.2. Adaptive Extended Kalman Filter.** Consider the nonlinear system given by (1) and (2); the AEKF used in this paper involves the following recursive relations:

$$\begin{aligned} \widehat{\mathbf{x}}_{k|k-1} &= \mathbf{A} \widehat{\mathbf{x}}_{k-1|k-1} \\ \mathbf{P}_{k|k-1} &= \mathbf{A} \mathbf{P}_{k-1} \mathbf{A}^T + \mathbf{Q} \\ \mathbf{K}_k &= \mathbf{P}_{k|k-1} (\mathbf{H}_k)^T \mathbf{S}_k^{-1} \\ \widehat{\mathbf{x}}_{k|k} &= \widehat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k \mathbf{r}_k \\ \mathbf{P}_{k|k} &= (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_{k|k-1} \\ \mathbf{r}_k &= \mathbf{z}_k - \mathbf{h}(\widehat{\mathbf{x}}_{k|k-1}) - \widehat{\mathbf{r}}_k \\ \mathbf{S}_k &= \mathbf{H}_k \mathbf{P}_{k|k-1} (\mathbf{H}_k)^T + \widehat{\mathbf{R}}_k. \end{aligned} \quad (4)$$

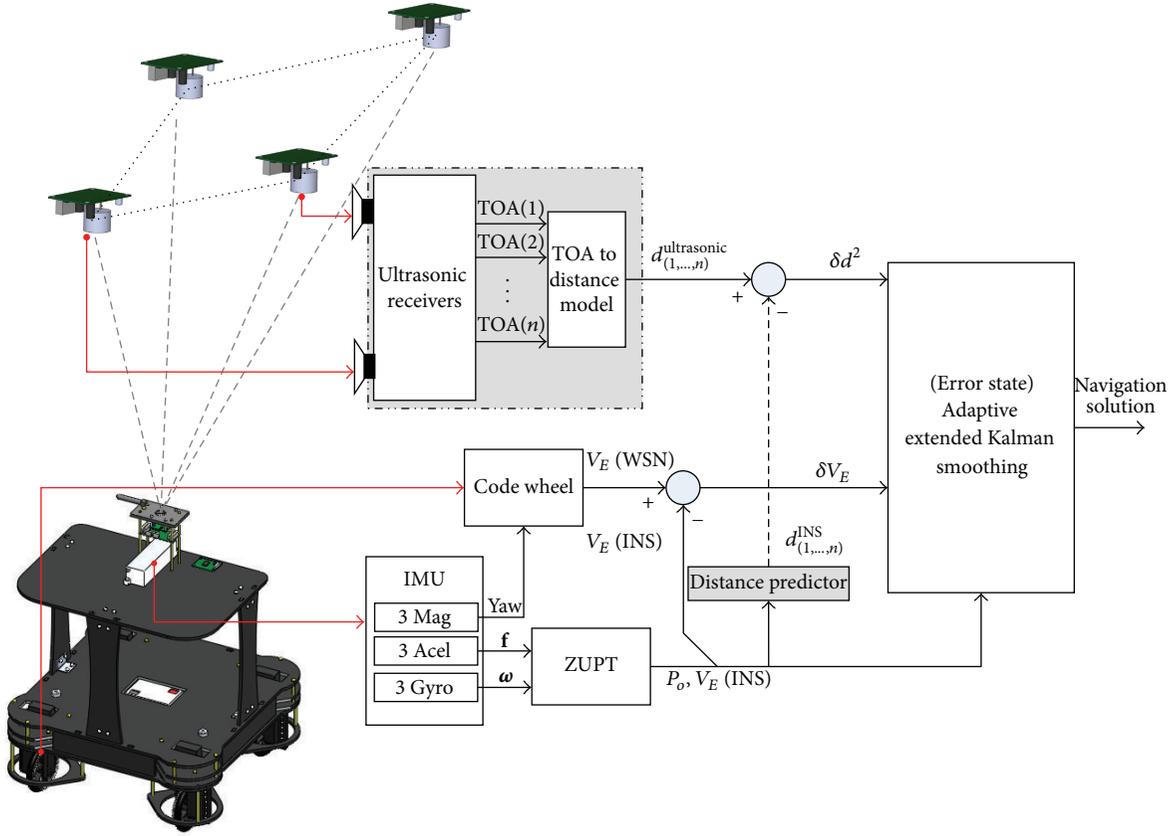


FIGURE 1: Configuration of the hybrid system.

Here,  $k = 1, 2, \dots, N$ ,  $\mathbf{H}(\widehat{\mathbf{X}}_{k|k}) = \partial \mathbf{h}(\widehat{\mathbf{X}}_{k|k}) / \partial \widehat{\mathbf{X}}_{k|k}$ ,  $\widehat{\mathbf{r}}_k$  and  $\widehat{\mathbf{R}}_k$  are computed by the time-varying noise statistics estimators with the following equations:

$$\widehat{\mathbf{r}}_k = (1 - d_{k-1})\widehat{\mathbf{r}}_{k-1} + d_{k-1}(\mathbf{z}_k - \mathbf{h}(\widehat{\mathbf{X}}_{k|k-1})) \quad (5)$$

$$\widehat{\mathbf{R}}_k = (1 - d_{k-1})\widehat{\mathbf{R}}_{k-1} + d_{k-1}(\mathbf{v}_k \mathbf{v}_k^T - \mathbf{H}_k \mathbf{P}_k (\mathbf{H}_k)^T)$$

Here,  $d_{k-1} = (1 - b)/(1 - b^k)$ ,  $0 < b < 1$ .

**2.3. Online Adaptive Extended Kalman Smoother.** In this work, in order to achieve high accuracy, adaptive two-extended-filter smoothing (ATEFS) is proposed. Consider the nonlinear system given by (1) and (2); the FKF employs the AEKF mentioned in Section 2.2, and the BKF is utilizing a set of equations as follows:

$$\mathbf{K}_k^B = \mathbf{P}_{k|k+1}^B (\mathbf{H}_k)^T (\mathbf{S}_k^B)^{-1}$$

$$\widehat{\mathbf{X}}_{k|k}^B = \widehat{\mathbf{X}}_{k|k+1}^B + \mathbf{K}_k^B \mathbf{r}_k^B$$

$$\mathbf{P}_{k|k}^B = (\mathbf{I} - \mathbf{K}_k^B \mathbf{H}_k) \mathbf{P}_{k|k+1}^B$$

$$\mathbf{r}_k^B = \mathbf{z}_k - \mathbf{h}(\widehat{\mathbf{X}}_{k|k+1}^B) - \widehat{\mathbf{r}}_k^B$$

$$\mathbf{S}_k = \mathbf{H}_k \mathbf{P}_{k|k+1} (\mathbf{H}_k)^T + \widehat{\mathbf{R}}_k$$

$$\widehat{\mathbf{X}}_{k-1|k}^B = \mathbf{A}^{-1} \widehat{\mathbf{X}}_{k|k}^B$$

$$\mathbf{P}_{k-1|k}^B = \mathbf{A}^{-1} \mathbf{P}_{k|k}^B (\mathbf{A}^{-1})^T + \mathbf{Q}. \quad (6)$$

Here,  $k = N - 1, N - 2, \dots, 1$ ,  $\mathbf{H}(\widehat{\mathbf{X}}_{k|k}) = \partial \mathbf{h}(\widehat{\mathbf{X}}_{k|k}) / \partial \widehat{\mathbf{X}}_{k|k}$ ,  $\widehat{\mathbf{r}}_k^B$  and  $\widehat{\mathbf{R}}_k^B$  are computed by the time-varying noise statistics estimators with the following equations:

$$\widehat{\mathbf{r}}_k = (1 - d_{k-1})\widehat{\mathbf{r}}_{k-1} + d_{k-1}(\mathbf{z}_k - \mathbf{h}(\widehat{\mathbf{X}}_{k|k+1}^B)) \quad (7)$$

$$\widehat{\mathbf{R}}_k = (1 - d_{k-1})\widehat{\mathbf{R}}_{k-1} + d_{k-1}(\mathbf{v}_k \mathbf{v}_k^T - \mathbf{H}_k \mathbf{P}_k (\mathbf{H}_k)^T),$$

where  $d_{k-1} = (1 - b)/(1 - b^k)$ ,  $0 < b < 1$ . Then, the smoothing estimate, that is, the combination of the FKF update and the BKF prediction, will be fixed as

$$\mathbf{K}_k^S = \mathbf{P}_{k|k+1}^B (\mathbf{P}_{k|k} + \mathbf{P}_{k|k+1}^B)^{-1}$$

$$\mathbf{X}_k^S = \mathbf{K}_k^S \widehat{\mathbf{X}}_{k|k} + (\mathbf{I} - \mathbf{K}_k^S) \widehat{\mathbf{X}}_{k|k+1}^B \quad (8)$$

$$\mathbf{P}_k^S = (\mathbf{P}_{k|k} + \mathbf{P}_{k|k+1}^B)^{-1}.$$

Here,  $k = 1, 2, \dots, N$ . Order for online smoothing, the AEKS and the average filter are used between two output

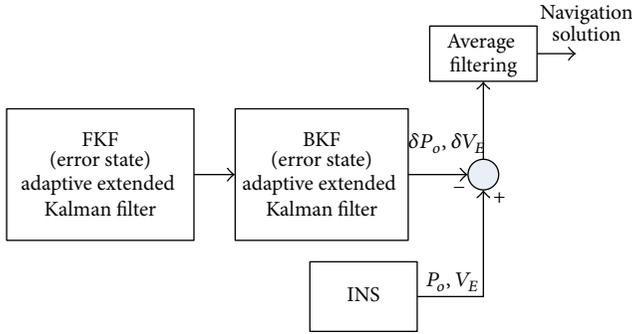


FIGURE 2: The process of online AEKS.



FIGURE 3: Real indoor test environment.

moments. When the integration filter needs to output the navigation solution, BKF is used to smooth the FKF output. The output of the BKF is used to compute the INS output estimation, and then the average of the INS output estimation between two output moments is sent as navigation solution. The process of online AEKS is shown in Figure 2.

### 3. Indoor Localization Tests and Performance

**3.1. Real Indoor Test Environment.** In order to assess the performance of the proposed method, two real indoor tests were done. The real indoor test environment is shown in Figure 3. In this work, one robot and 6 RNs are employed for the test. Both the robot and the RN are marked in Figure 3. As shown in Figure 4, the robot is the carrier of the IMU and the ultrasonic sender. It is able to collect the data of IMU and the distances between the robot and the RNs by using the PC fixed on the robot. In this work, the RN is used to receive the signal of the ultrasonic ranging sent by the ultrasonic sender and calculate the distance between the RNs and robot. It is also able to send the sensor data to the ultrasonic sender when

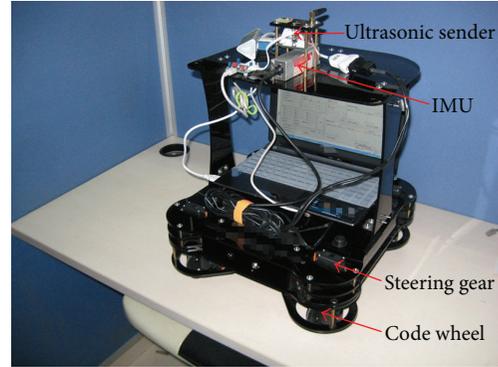


FIGURE 4: The prototype of the robot.

it gets the command. The sample time used in this work is 0.02 s.

Figure 5 displays the trajectories of the real tests. The robot runs from the beginning point (denoted by a black square) to the end point (denoted by a black circle) with 0.33 m/s. Meanwhile, the RNs are denoted by yellow circles in Figure 5.

**3.2. The Position Errors of the Proposed Method.** In this section, the position errors are discussed. The position errors for the AEKF, average filter of AEKF (average AEKF), and the proposed method are shown in Figure 6. In the figures, the AEKF solution is depicted in blue, the green line represents the average AEKF solution, and the proposed method employs the green line.

The east and north position errors of the first trajectory are shown in Figures 6(a) and 6(b), respectively. In Figure 6(a)(A), it can be seen that the AEKF is able to keep the east position error about 0.0400 m, and it decreases the mean east position errors by about 58.30% and 82.63% compared with least squares (LS) solution and the INS-only solution, respectively. The average AEKF outputs the average value of the AEKF solution between two output moments, and the results show that it is able to reduce the mean east position errors to 0.0389 m. Figure 6(a)(B) shows the east position errors for the average AEKF solution and the proposed method. It is easy to see that the proposed method is effective to reduce the east position error, and the results show that the mean east position of the proposed method is 0.0370 m. To the north position errors of the first trajectory, from Figure 6(b), it is easy to see that the proposed method solution also has the lowest error. The mean north position of the proposed method is 0.0262 m, and it reduces the mean north position errors by about 5.2% and 1% compared with the AEKF solution and the average AEKF solution.

Figures 6(c) and 6(d) show the east and north position errors of the second trajectory, respectively. In Figure 6(c), it is easy to see that the proposed method solution also has the lowest error. The mean east position of the proposed method is 0.0253 m, and it reduces the mean north position errors by about 25.56% and 14.08% compared with the AEKF solution and the average AEKF solution. The north position errors of the second trajectory are shown in Figure 6(d), and similar

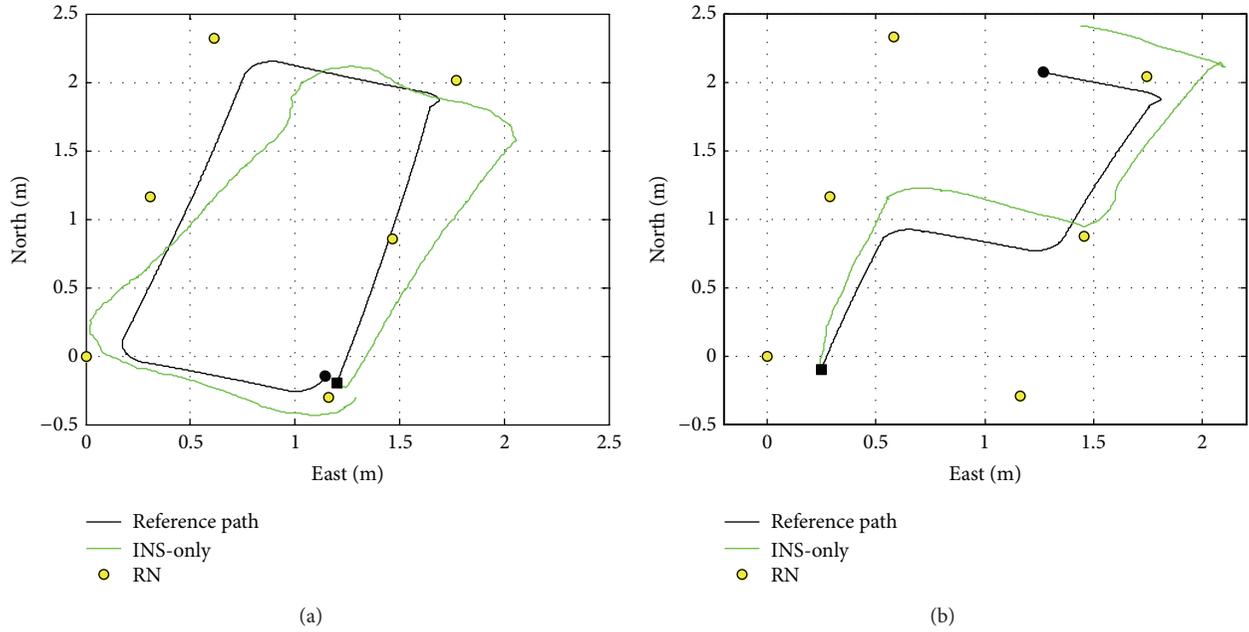


FIGURE 5: The trajectory of the real test.

TABLE 1: The position errors of the trajectories.

	The first trajectory		The second trajectory	
	Mean east position error (m)	Mean north position error (m)	Mean east position error (m)	Mean north position error (m)
INS-only	0.2308	0.1412	0.1270	0.2418
LS	0.0961	0.0616	0.0808	0.0557
AEKF	0.0401	0.0276	0.0340	0.0310
off-line AEKS	0.0362	0.0239	0.0273	0.0241
Average AEKF	0.0389	0.0264	0.0294	0.0253
The proposed method	0.0370	0.0262	0.0253	0.0218

TABLE 2: The velocity errors of the trajectories.

	The first trajectory		The second trajectory	
	Mean east velocity error (m/s)	Mean north velocity error (m/s)	Mean east velocity error (m/s)	Mean north velocity error (m/s)
INS-only	0.0868	0.0847	0.0768	0.0788
LS	0.0368	0.0627	0.0470	0.0541
AEKF	0.0329	0.0436	0.0382	0.0391
off-line AEKS	0.0308	0.0371	0.0275	0.0297
Average AEKF	0.0320	0.0438	0.0421	0.0407
The proposed method	0.0273	0.0361	0.0274	0.0267

to the first trajectory, the average AEKF solution reduces the position error from 0.0310 m measured by AEKF to 0.0253 m; then the proposed method improves the accuracy to 0.0218, and it reduce the mean north position errors by about 13.87%. The proposed method used in the second trajectory is significantly effective than that used in the first trajectory.

The performance for the INS-only, LS, AEKF, offline AEKS, average AEKF, and the proposed method is shown in Table 1.

3.3. *The Velocity Errors of the Proposed Method.* The velocity errors are discussed in this section. Figure 7 displays the position errors for the AEKF, average filter of AEKF (average

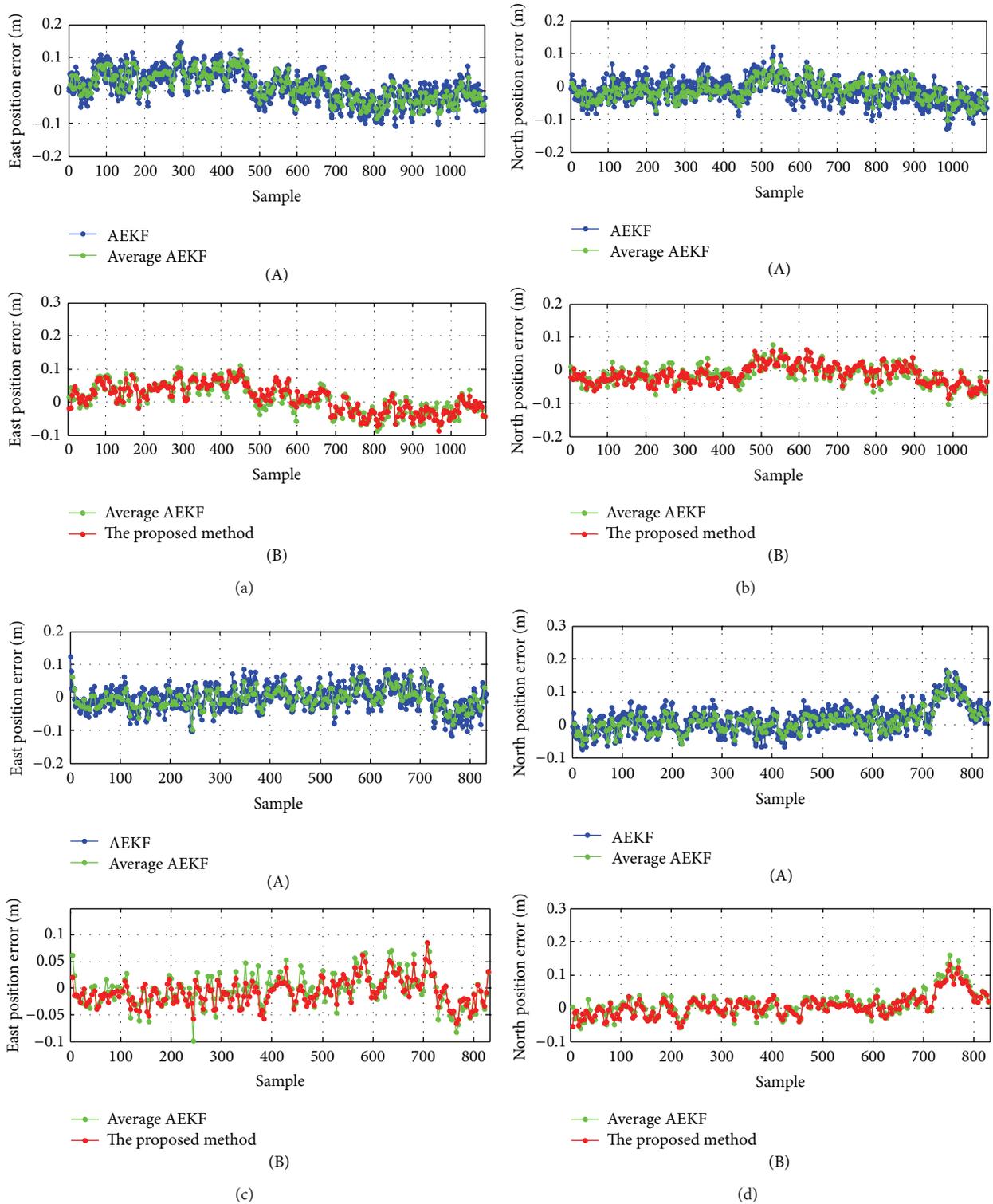


FIGURE 6: The position errors for the AEKF, average AEKF, and the proposed method.

AEKF), and the proposed method. Similar to Figure 6, the AEKF solution is depicted in blue, the green line represents the average AEKF solution, and the proposed method employs the green line.

Figures 7(a) and 7(b) display the east and north velocity errors of the first trajectory, respectively. In Figure 7(a), it can be seen that the proposed method has the lowest error. The results show that it decreases the mean east position errors by

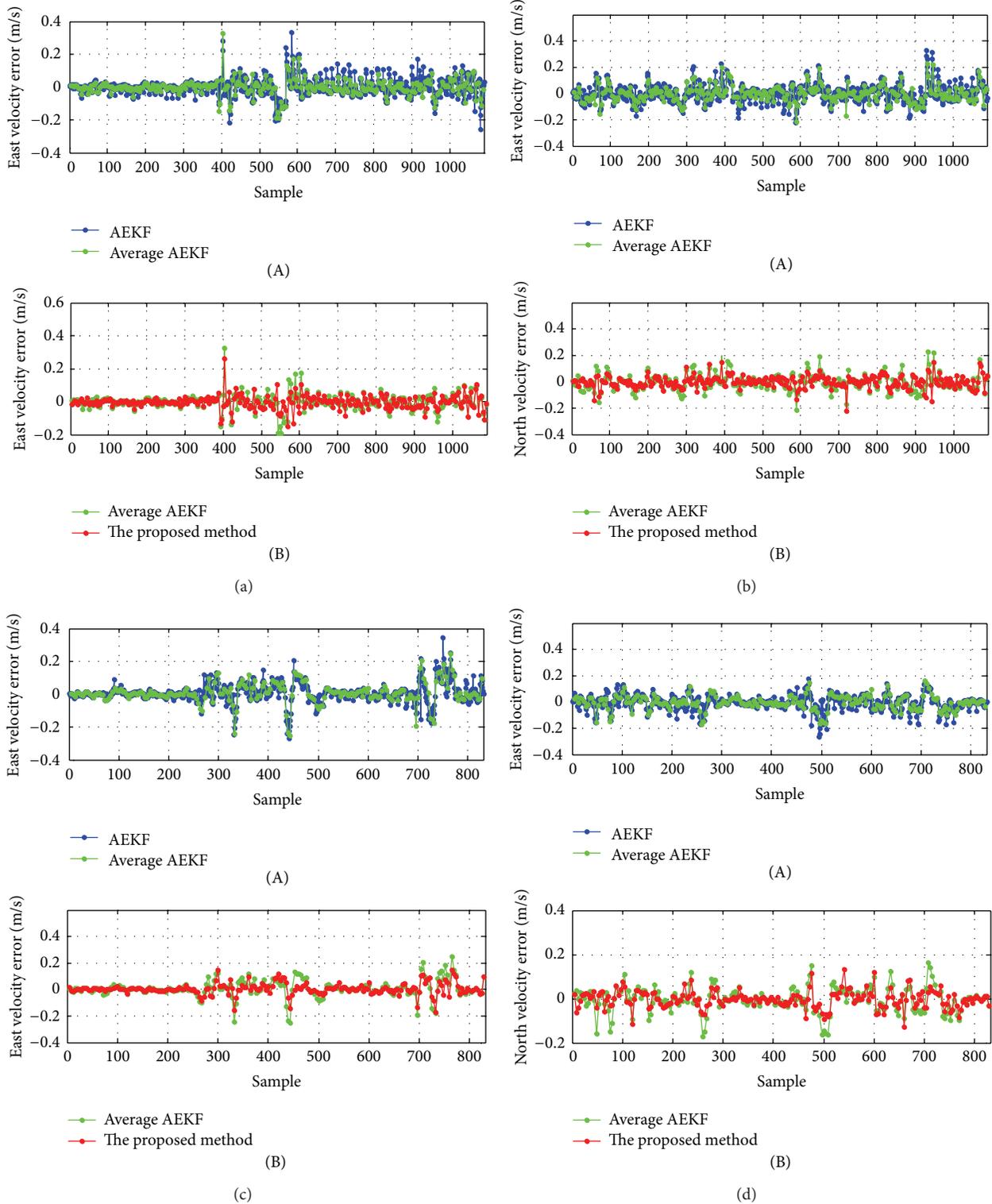


FIGURE 7: The velocity errors for the AEKF, average AEKF, and the proposed method.

about 27.74% and 62.06% compared with LS solution and the INS-only solution, respectively. To the north velocity errors of the first trajectory, from the Figure 7(b)(A), it can be seen that the average AEKF is more effective than the AEKF, and the results show that the mean north velocity of the average

AEKF is 0.0436 m/s. Figure 7(b)(B) shows the north velocity errors for the average AEKF and the proposed method, it can be seen that the proposed method decreases the mean north position errors from 0.0436 m/s to 0.0361 m/s. The east and north velocity errors of the second trajectory are shown

in Figures 7(c) and 7(d), respectively. From the figures, it can be seen that the proposed method has the lowest error, and the mean velocity errors of the second trajectory in east direction and north direction is 0.0274 m/s and 0.0267 m/s. The performance for the INS-only, LS, AEKF, offline AEKS, average AEKF, and the proposed method is shown in Table 2.

#### 4. Conclusions

This work proposed the design and implementation of AEKS on INS/WSN integration system for mobile robot indoors. In this mode, the AEKF is employed to improve the forward filtering output accuracy, and the back filter is used to smooth the forward filtering output. In order to achieve online smoothing, the AEKS and the average filter are used between two output period. Two real indoor tests have been done to assess the performance of the proposed method, and the experimental results show that proposed method is the most effective method to estimate the navigation information and give the optimal state estimation.

#### Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (nos. 51375087, 41204025, 50975049), Ocean Special Funds for Scientific Research on Public Causes (no. 201205035-09), Specialized Research Fund for the Doctoral Program of Higher Education (no. 20110092110039), the 52th China Postdoctoral Science Foundation (no. 2012M520967), and the Program Sponsored for Scientific Innovation Research of College Graduate in Jiangsu Province, China (no. CXLX\_0101).

#### References

- [1] S. J. Kim and B. K. Kim, "Accurate hybrid global self-localization algorithm for indoor mobile robots with two-dimensional isotropic ultrasonic receivers," *IEEE Transactions on Instrumentation and Measurement*, vol. 60, no. 10, pp. 3391–3404, 2011.
- [2] A. R. J. Ruiz, F. S. Granja, J. C. P. Honorato, and J. I. G. Rosas, "Accurate pedestrian indoor navigation by tightly coupling foot-mounted IMU and RFID measurements," *IEEE Transactions on Instrumentation and Measurement*, vol. 61, no. 1, pp. 178–189, 2012.
- [3] G. Dedes and A. Dempster, "Indoor GPS positioning-challenges and opportunities," in *Proceedings of the IEEE Vehicular Technology Conference (VTC '05)*, pp. 412–415, 2005.
- [4] A. A. N. Shirehjini, A. Yassine, and S. Shirmohammadi, "An RFID-based position and orientation measurement system for mobile objects in intelligent environments," *IEEE Transactions on Instrumentation and Measurement*, vol. 61, no. 6, pp. 1664–1675, 2012.
- [5] W.-C. Park and M.-H. Yoon, "The implementation of indoor location system to control ZigBee home network," in *Proceedings of the SICE-ICASE International Joint Conference*, pp. 2158–2161, Busan, Republic of Korea, October 2006.
- [6] M. M. Saad, C. J. Bleakley, T. Ballal, and S. Dobson, "High-accuracy reference-free ultrasonic location estimation," *IEEE Transactions on Instrumentation and Measurement*, vol. 61, no. 6, pp. 1561–1570, 2012.
- [7] L. Fang, P. Antsaklis, L. Montestruque et al., "Design of a wireless assisted pedestrian dead reckoning system—the NavMote experience," *IEEE Transactions on Instrumentation and Measurement*, vol. 54, no. 6, pp. 2342–2358, 2005.
- [8] Z.-K. Xu, Y. Li, C. Rizos, and X. Xu, "Novel hybrid of LS-SVM and kalman filter for GPS/INS integration," *Journal of Navigation*, vol. 63, no. 2, pp. 289–299, 2010.
- [9] J. Lobo, P. Lucas, J. Dias, and A. Traca de Almeida, "Inertia navigation system for mobile land vehicles," in *Proceedings of the IEEE International Symposium on Industrial Electronics (ISIE '95)*, pp. 843–848, July 1995.
- [10] N. El-Sheimy and K. P. Schwarz, "Integrating differential GPS receivers with an inertial navigation system (INS) and CCD cameras for a mobile GIS data collection system," in *Proceedings of the International Society for Photogrammetry and Remote Sensing Conference*, pp. 241–248, Ottawa, Canada, 1994.
- [11] F. Chen and M. W. Dunnigan, "Comparative study of a sliding-mode observer and kalman filters for full state estimation in an induction machine," *IEE Proceedings, Electric Power Applications*, vol. 149, no. 1, pp. 53–64, 2002.
- [12] H. Shao, D. Kim, and K. You, "TDOA/FDOA geolocation with adaptive extended kalman filter," *Communications in Computer and Information Science*, vol. 121, pp. 226–235, 2010.
- [13] K.-W. Chiang, T. T. Duong, J. K. Liao et al., "On-line smoothing for an integrated navigation system with low-cost MEMS inertial sensors," *Sensors*, vol. 2012, pp. 17372–17389, 2012.
- [14] H. Rauch, F. Tung, and C. Striebel, "Maximum likelihood estimates of linear dynamic systems," *AIAA Journal*, vol. 3, no. 8, pp. 1445–1450, 1965.
- [15] H. Liu, S. Nassar, and N. El-Sheimy, "Two-filter smoothing for accurate INS/GPS land-vehicle navigation in urban centers," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 9, pp. 4256–4267, 2010.
- [16] Y. Xu, X.-Y. Chen, and Q.-H. Li, "Unbiased tightly-coupled INS/WSN integrated navigation based on extended kalman filter," *Journal of Chinese Inertial Technology*, vol. 20, no. 3, pp. 292–299, 2012.

## Research Article

# Entropy-Based and Weighted Selective SIFT Clustering as an Energy Aware Framework for Supervised Visual Recognition of Man-Made Structures

**Ayman El Mobacher, Nicholas Mitri, and Mariette Awad**

*Electrical and Computer Engineering Department, The American University of Beirut, P.O. Box 11-0236, Riad El Solh, Beirut 1107 2020, Lebanon*

Correspondence should be addressed to Mariette Awad; [mariette.awad@aub.edu.lb](mailto:mariette.awad@aub.edu.lb)

Received 14 July 2013; Accepted 30 September 2013

Academic Editor: Yudong Zhang

Copyright © 2013 Ayman El Mobacher et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Using local invariant features has been proven by published literature to be powerful for image processing and pattern recognition tasks. However, in energy aware environments, these invariant features would not scale easily because of their computational requirements. Motivated to find an efficient building recognition algorithm based on scale invariant feature transform (SIFT) keypoints, we present in this paper uSee, a supervised learning framework which exploits the symmetrical and repetitive structural patterns in buildings to identify subsets of relevant clusters formed by these keypoints. Once an image is captured by a smart phone, uSee preprocesses it using variations in gradient angle- and entropy-based measures before extracting the building signature and comparing its representative SIFT keypoints against a repository of building images. Experimental results on 2 different databases confirm the effectiveness of uSee in delivering, at a greatly reduced computational cost, the high matching scores for building recognition that local descriptors can achieve. With only 14.3% of image SIFT keypoints, uSee exceeded prior literature results by achieving an accuracy of 99.1% on the Zurich Building Database with no manual rotation; thus saving significantly on the computational requirements of the task at hand.

## 1. Introduction

Given the impressive proliferation of digital images and videos nowadays which is partly caused by the ease of acquiring them using smartphones, opportunities for novel visual mining and search applications based on image processing are emerging as hot topics for further exploration. However, because of the energy aware computing trends and the somewhat still limited processing capabilities of these handheld devices, the traditional image and video processing techniques available would require adaptation to better fit an environment where “green”, “mobility”, and “on the go” are becoming prevailing.

We present, in this paper, uSee, a supervised learning framework designed for exploitation of the physical world and delivery of context-based services after an efficient usage of scale invariant feature transform (SIFT) keypoints that have been widely and successfully applied for processing

and mining images. Different from the means available for obtaining the absolute/relative location which provide a lot of information in a geographic sense, but modestly with regards to context, uSee is implemented as an on-demand pull service based on energy aware processing of building images. Any authorized user with a handheld device having both a camera and either an internet connection or MMS settings could take a picture of a residential, commercial building, or a landmark and receive from uSee, in near real-time, touristic, advertisement, or entertainment information specific to the building of interest. From an application perspective, uSee is intended to be customized and maintained for specific cities. It would help improve touristic experience, promote visually triggered marketing methods, and even assist in ubiquitous learning as people would relate better to their environment because of the information it provides. From a workflow perspective, and to the best of our knowledge, uSee's novelty relies in its computational phases. Different

from the prior work, it employs an entropy-based metric and exploits repetitive patterns and symmetry in man-made structures to identify a signature of selected SIFT keypoints representative of the building in question using correlation and weighted clustering measures. Experimental results showed noticeable decrease in computational requirements while maintaining minimal performance degradation when compared to building identification based on a full utilization of SIFT keypoints. The remainder of the paper is organized as follows: Section 2 surveys prior work related to building and landmark recognition followed by a detailed description in Section 3 of uSee entropy-based segmentation, signature creation, and building identification phases. Experimental results are discussed in Section 4 before concluding remarks and planned research extensions are presented in Section 5.

## 2. Related Work

Lately, similar applications to building recognition approached the problem of effective image search mainly on landmarks. Kennedy and Naaman in [1] developed a system that generates representative images for landmarks based on community-contributed collections of images. Using tags and locations to cluster landmarks and some visual features (color and texture as global features and SIFT as local ones) of the clustered images, they selected the images with the highest connections to other images in their group as representative images. Quack et al. [2], instead, exhaustively gathered geo-tagged data by first dividing the earth's surface into tiles. They then downloaded the geo-tagged photos related to each tile along with their tags and timestamps and used text and visual features as well as speeded up robust features (SURF) to group photos within a given tile. After the clustering stage, they proceeded to automatically annotate the clusters and finally, they linked them to related articles on the internet. In [3] Zheng et al. used travel guide articles in addition to geo-tagged images, hence, combining set of images from 2 distinct sources to mine for landmarks. They, next, clustered the images gathered based on direct keypoint matching. Although [1–3] used SIFT keypoints, none of them suggested to improve on the computational requirements in these image processing problems by reducing operational complexity.

From a general classification perspective with respect to building recognition algorithms, the related work available in the literature is also different from the proposed uSee. While earlier efforts relied on combinations of global features and attempts to define effective local features, the most recent research focused mostly on invariant local features. Pass and Zabih in [4] used a joint histogram approach that combines global features such as color, edge density, texture, gradient magnitude, and rank features. In [5], Shao et al. achieved a recognition rate of 94.8% on the Zurich Building Database (ZuBuD) [6] by using scale invariant descriptors followed by a nearest neighbor search in the database for the best match based on “hyper polyhedron with adaptive threshold” indexing. Zhang and Kosecka in [7, 8] opted for a 2-stage-approach in their recognition phase. Selecting first the top candidates

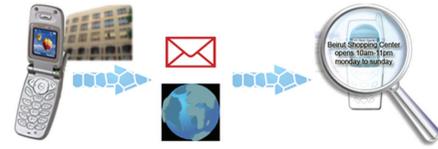


FIGURE 1: uSee pictorial workflow.

based on a localized color histogram that classifies gradient angles to the dominant vanishing points in the image, they applied SIFT features to the top candidates to detect the best match. Although they achieved 96.5% overall recognition rate, they had to manually prerotate some of the query images. Furthermore, the query images of the ZuBuD database do not pose major changes in views or illumination. Authors in [9] applied a global set of features invariant to illumination and clutter changes to extract the best candidates, thus, reducing the search space and then matched in a second stage exhaustively all keypoints. However, their global features performed poorly when pose, scale, and rotation changes were involved for selecting the best candidates of a match. The works in [10, 11] proposed using visual salient regions and attention model-based filtration so that only keypoints within the region of interest are used in the matching process while dropping those in the background. In building recognition, the building itself will often cover most of the image, and thus, a salient region will not help reduce much the amount of keypoints. Our previous work on HISI [12] proposed a soft recognition approach to the processing and identification of ZuBuB buildings. Using a coarse joint histogram technique, an image captured by a mobile user with a cell phone is preprocessed to reduce the search space to an adaptive list of potential buildings, after which a weighted fusion of different SIFT maps identifies the building in question.

## 3. uSee Preprocessing and Identification

uSee is a supervised framework for recognizing building in an energy aware fashion that will minimize the resource requirements associated with this task. Its preprocessing and identification phases could be performed either in the cloud or locally by the mobile application. In essence, the handheld device would either:

- (a) Perform all the pre-processing tasks and update the database if needed with new buildings and/or context aware services provided by different users. This would require the device to have enough computing power and memory to store the buildings' database and an Internet connection;
- (b) send the image to the cloud for pre-processing as shown in Figure 1 which represents the current implementation of uSee. The remote server is expected to identify the building and notify the user by SMS or email with the available location-based information.

**3.1. Entropy-Based Segmentation.** uSee image preprocessing phase starts with generating a map that highlights the areas



FIGURE 2: (a) Original image. (b) Entropy based preprocessed image.

of the image with high variation in gradient angle. This is done by checking the variation of the gradient angles in a given window's rows and columns. After classifying the gradient angles into 9 values equally spaced on a semicircle, we convolve the gradient angle image with a kernel that will compute a product of the entropy of the gradient angles in the kernel rows and columns based on the following.

- (i) The histogram along the rows of the window is formed of size  $9 \times L$ , where 9 is the number of classes of gradient angles on the semicircle and  $L$  is the size of the window around the given pixel.
- (ii) Based on the number of votes in the histogram defined above, a discrete probability distribution is formed for each of the  $L$  rows of the window.
- (iii) Steps 1 and 2 are repeated similarly for the columns of the window.
- (iv) Once the column and row discrete probability distributions are formed, the row and column entropies are computed in a straightforward manner as shown in (1) below:

$$\text{entropy}_R = \begin{bmatrix} -(p_{\text{row},R-(L-1)/2}) \cdot \ln(p_{\text{row},R-(L-1)/2})^T \\ \vdots \\ -(p_{\text{row},R+(L-1)/2}) \cdot \ln(p_{\text{row},R+(L-1)/2})^T \end{bmatrix}, \quad (1)$$

$$\text{entropy}_C = \begin{bmatrix} -(p_{\text{col},C-(L-1)/2}) \cdot \ln(p_{\text{col},C-(L-1)/2})^T \cdots \\ \vdots \\ -(p_{\text{col},C+(L-1)/2}) \cdot \ln(p_{\text{col},C+(L-1)/2})^T \end{bmatrix},$$

where  $R$  and  $C$  are the row and column of the pixel whose noise level is to be calculated,  $L$  is the size of the kernel, and  $p_{\text{row},i}$  and  $p_{\text{col},j}$  are the discrete probability distributions of gradient angles along row  $i$  and column  $j$ , respectively, of the kernel.

The product of the exponential of the 2 vectors  $\text{entropy}_C$  and  $\text{entropy}_R$  gives the pixel noise level as in (2) where the noise at pixel of row  $R$  and column  $C$  is incremented by the product of the exponential of the entropies. The exponential

is used to counter the logarithmic effect in the entropy as follows:

$$\text{noise}_{R,C} = \text{noise}_{R,C} + (e^{\text{entropy}_C} \times e^{\text{entropy}_R}). \quad (2)$$

The image is then divided using k-means into 2 clusters based on the pixel noise level, one with a low gradient angle variation and the other with a high gradient angle variation. The high dispersion in a window is interpreted as indicative of the presence of omnidirectional edges which are not characteristic of building edges, instead they could be the most nonstructural patches in the image such as trees, branches and bushes. Figure 2 illustrates the output of the gradient angle variation classifier using the entropy measure defined above. The SIFT keypoints of the image that are extracted and that lie in the high gradient angle variation region are eventually filtered out.

**3.2. SIFT Signature Extraction.** uSee exploits the embedded symmetry and repetitive patterns in man-made structures to ensure an energy aware framework. To select the most relevant keypoints, the autocorrelation matrix of the image's keypoints is generated by directly computing the inner product of the list of keypoints since they are of unit norm as shown:

$$C = [k_1 \cdots k_n]^T \cdot [k_1 \cdots k_n], \quad (3)$$

where  $C$  is the correlation matrix and  $k_i$  is the 128-dimensional feature vector of SIFT keypoint  $i$  for an image with  $n$  SIFT keypoints.  $C$  is then thresholded at  $\text{Th}$ , which is determined experimentally, and correlation clustering is then performed. uSee workflow, as shown in Figure 4, proceeds as follows.

- (i) Step 1: The SIFT keypoint  $b$  with the highest number of correlated keypoints in matrix  $C$  is selected as the head of the current cluster  $cc$ , and all  $m$  keypoints correlated to it above the predefined threshold  $\text{Th}$  will constitute the cluster samples of cluster  $i$  denoted as  $s_i$ .
- (ii) Step 2: All elements of cluster  $s_i$  are removed from  $n$ , the original set of SIFT keypoints, and the next cluster is formed similar to Step 1, with the remaining  $(n-m-1)$  SIFT keypoints.

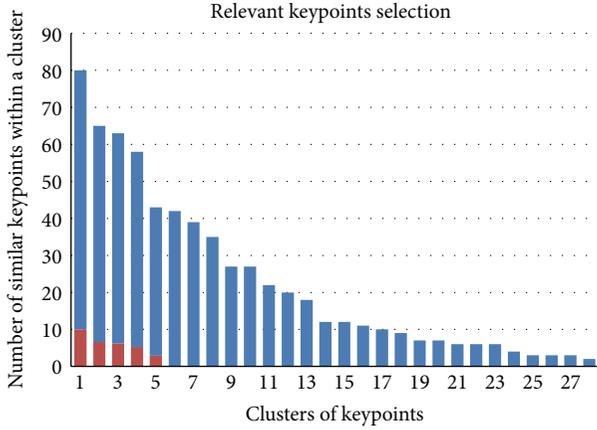


FIGURE 3: Graph of relevant keypoints selection.

- (iii) Step 3: Steps 1 and 2 are repeated until all  $n$  keypoints are clustered into  $c$  clusters.
- (iv) Step 4: The  $r$  most representative SIFT keypoints are selected based on a weighted score so as to avoid any bias towards one specific cluster of keypoints.  $r$  is predefined for any given image. In our experiments, as we shall see in Section 4, we tried out different values for  $r$  that ranged from 2.7% to 15.5% of the total  $n$  SIFT keypoints. Figure 3 shows the relevant keypoints selected by cluster in comparison to the total keypoints. Each bar represents a cluster of similar keypoints; the blue bars represent the total number of keypoints in a given cluster, and the red bars are the relevant keypoints selected within the given cluster. The ratio of the red bar to the blue one is the same as that of the blue one to the total keypoints of the image.

Note that selection of keypoints (red bars) stops when the specified limit number  $r$  is reached which, in the graph above, happens at the 6th cluster.

Each cluster  $s_i$  contributes in points to the overall signature proportionally to its cluster size such that  $\sum_{i=1}^c s_i = r$  is reached, where  $s_i$  is the set of selected keypoints within a cluster (whose size corresponds to each red bar in the graph of Figure 3) and  $r$  is the limit chosen as the maximum number of selected keypoints. This guarantees a more diverse and balanced selection of the most highly correlated keypoints and will form the signature of the building.

While Figure 5 demonstrates how the SIFT keypoints of an image are reduced by uSee selection and clustering process, Figure 6 illustrates the weighting score selection mechanism of Step 4. In the left image, Figure 6 shows that  $r$  was reached within only one cluster whereas in the image on the right  $r$  is formed of SIFT keypoints from five different clusters. The map of Figure 7 clearly highlights the repetitive pattern of the corners in windows and shows where brighter dots reflect higher number of similar keypoints.

**3.3. uSee Identification.** When a new image is acquired, its signature is extracted and the L2 norm is computed between

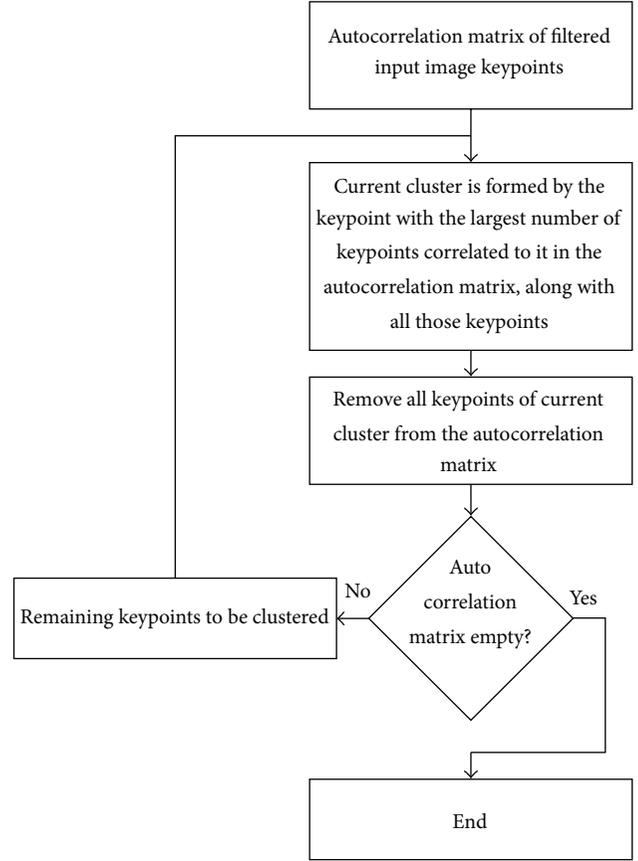


FIGURE 4: uSee SIFT keypoints clustering workflow.

its signature and the database's keypoints, or equivalently, the inner product can be computed then instead of selecting the lowest distance; the highest cosine angle is used. Building identification is based on a decision workflow that relies on a maximum voting scheme as shown below.

Assume that.

$k_q$  is the keypoints matrix of size  $[128 \times r]$  of the image to be classified,

$k_d$  is the keypoints matrix of the database which is the concatenation of all keypoints matrices of all the database images; thus, it is of size  $[128 \times (r \cdot d)]$  where  $d$  is the number of images in the database.

Then the correlation matrix formed by the inner product of  $k_q$  and  $k_d$  is given thus by  $C_{q,d} = k_q^T \times k_d$ , which is a matrix of size  $[r \times (r \cdot d)]$ .

For a given row  $i$  of  $C_{q,d}$ , the values in the columns represent the correlation between the database keypoints and the query's  $i$ th keypoint. The highest value in the  $i$ th row is the closest keypoint match. The corresponding image of the given column with highest value in row  $i$  gets a vote. Equation (4) maps the given column to its corresponding image.

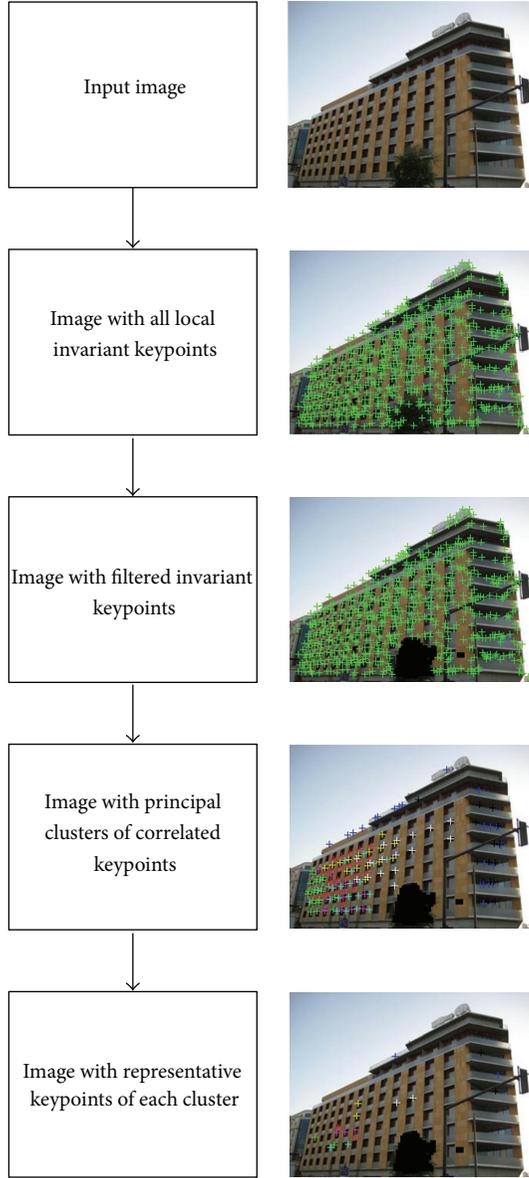


FIGURE 5: uSee selection workflow of relevant SIFT keypoints.

All in all, there will be  $r$  votes for a given query image, and the best match will be the image with highest votes as in (4) and (5) as follows:

$$k = \text{floor} \left( \frac{\max_j (C_{q,d}(i, j)) - 1}{r} \right) + 1, \quad (4)$$

$$M = \max_k V_m, \quad (5)$$

where  $V_m$  is the matching score vector of size  $1 \times d$  ( $d$  is the total number of images in the database) in which the  $k$ th element of  $V_m$  is the number of votes that image  $k$  of the database received, and the best match  $M$  in (5) is the index of  $V_m$  corresponding to its maximum value. With the current

uSee implementation, once a building is identified, context-related information stored in the database will be retrieved and sent to the user.

## 4. Experimental Results

**4.1. ZuBuD Database.** The first experiments were run on ZuBuD, a popular benchmark used in computer vision and which has a total of 1005 database images captured at a resolution of  $480 \times 640$  and 115 query images downsized to  $240 \times 320$ . To be able to compare with published results, we tested uSee on ZuBuD based on the query set and reported recognition accuracy as the average for the 201 buildings. Several different values for  $r$  were tested for both the reference and the query images. The average number of all SIFT keypoints based on the implementation done in [13] in a given image was observed to be 740. Table 1 shows a summary of the results for  $\text{Th} = 0.9$ . It can be seen from the 3rd row of Table 1 that with a selection as low as 20 relevant keypoints (about 2.7% of the total SIFT keypoints), the recognition rate is above 90%. With  $r = 50$  (6.8%), the recognition rate already matches the best result reported in the literature at 96.5% [7, 8]. The remarkable efficiency of uSee can be best demonstrated by this result where the computational complexity is reduced; thus, energy minimization is maximized compared to the case when all SIFT keypoints are used in building identification.

Increasing  $r$  to only 14.3% of the total SIFT keypoints exceeded all prior results achieved, to the best of our knowledge, on ZuBuD and reached 99.1% accuracy in building recognition. Also, uSee did not require any manual rotation for any image as was done in prior research work to be able to correctly identify rotated building images. The only image that failed in this case, as shown in Figure 8, had clutter that resulted after segmentation, with a small area of the image for building recognition.

The reduction in operational complexity that uSee allows for is substantial. When a query image is presented to uSee, the correlation matrix is computed then used to compare the query keypoints against all database keypoints before tallying the votes. With all  $n$  keypoints used, the complexity of creating the matrix and finding the winners is  $O(n^2)$ . Using only a subset  $r$  as suggested in uSee, the ratio of computational costs is  $(r/n)^2$ . For instance, with 50 keypoints in the query image ( $r = 6.8\%$ ), we save 99.54% on computing energy without affecting accuracy results. The last column in Table 1 represents the percentage of the computational energy computed as the ratio of the proposed SIFT keypoints matching computation to the total keypoints comparison, that is, when no SIFT keypoint reduction is performed. As can be seen, the savings are significant and promising.

**4.2. Beirut Database.** Because uSee is meant to be used as a real-life application, we decided to evaluate its performance on a more challenging set of images with more severe illumination, pose, and scale changes. Therefore, further tests were conducted on another database of buildings from the city of Beirut. The Beirut database was formed with



FIGURE 6: Selection of relevant keypoints based on one cluster (a) and 5 different clusters for better diversity in selection (b).

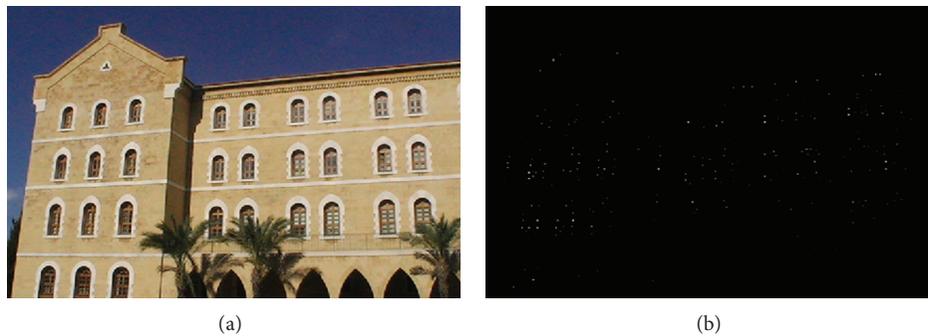


FIGURE 7: (a) Original image. (b) Map of recurrent keypoints; the brighter the dots the more similar keypoints there are to the dot.



FIGURE 8: Failed image in ZuBuD database with  $r = 100$  keypoints.

5 reference images taken at the same time of day and a query image at different times and weather conditions. This means that the reference images set is more homogeneous in terms of illumination, but the test images are different from their corresponding reference in illumination, camera angle, and scale which are major image processing challenges not present in the ZuBuD database. All Beirut images were taken at a resolution of  $480 \times 640$  and downsized to  $240 \times 320$ . The Beirut database consists of a total of 38 buildings from downtown Beirut and from the American University of Beirut to which some context aware information was added. In order to compare uSee accuracy, we ran some live test using

a BlackBerry phone using the same approach adopted with ZuBuD, which means that the recognition rate reported is the average value computed after testing the query images for all 38 buildings.

To rule out the chance that the choice of  $Th$  is adversely affecting performance, we investigated the impact of different correlation threshold on the Beirut database. Table 2 shows the recognition rates achieved at varying values of  $Th$ . At lower thresholds, the algorithm will cluster together keypoints that are poorly correlated. At higher thresholds, it will filter out too many potential candidates of a common cluster. Hence, two similar keypoints will end up being considered different, thus, leading to poor diversity in keypoints selection for the image signature. The tests show that a value between 0.88 and 0.9 yields good results, and more specifically, 0.88 as a correlation threshold yields the best results and is optimal for the Beirut database. This threshold value results in an 84.93% recognition rate and is used for the remainder of this section.

We tested uSee performance against a random selection of SIFT keypoints. The left plot in Figure 9 shows the recognition curve plotted against the number of keypoints selected for a  $Th = 0.88$ . A log-scale is used for the horizontal axis to enhance the view at the lower end of the keypoints selection. The red curve shows the performance of the randomly selected keypoints, and the blue curve depicts the  $r$  SIFT keypoints selected by uSee. It can be seen as well

TABLE I: Summary of results for the ZuBuD database.

Method	Keypoints in reference images	Keypoints in query image	Ratio of selected keypoints to total keypoints	Recognition rate	Ratio of computational energy costs
Work in [2]	All in filtered region	All in filtered region	—	94.80%	n/a
Work in [1]	All in filtered region	All in filtered region	—	96.50%	0.52%
Relevant keypoints	20	20	2.7%	91.30%	0.07%
	30	30	4.1%	94.80%	0.16%
	40	40	5.4%	95.70%	0.29%
	50	50	6.8%	96.50%	0.46%
	100	75	10.1%	98.30%	1.37%
	100	115	15.5%	99.10%	2.10%

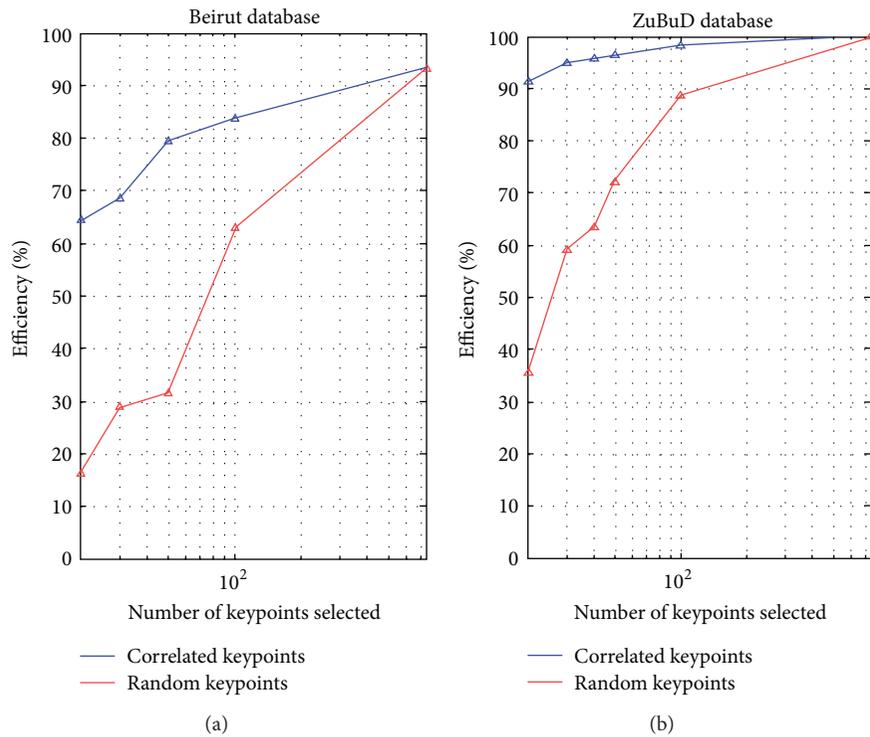


FIGURE 9: Recognition curve for Beirut (a) and ZuBuD (b) databases.

as intuitively understood that increasing  $r$  will have both curves converge towards each other and towards the maximum performance, that is, that of selecting all  $n$  keypoints. However, the more interesting part of the curves occurs at the lower end and specifically between 20 and 50 keypoints since 50 keypoints achieved the same best result reported by [7, 8] in the literature for the ZuBuD database. The bigger discrepancies between the 2 curves in the Beirut database is an indication of the higher complexity of that database, as is the fact that when all  $n$  keypoints are used, a maximum of only 93.2% accuracy was achieved.

Samples of buildings that failed to be identified by uSee in Beirut database are shown in Figure 10, while sample of the query and reference images for the ZuBuD database are displayed in Figure 11. To the left of the green line in Figure 11 is the query image that has been successfully recognized, and to its right are the reference images in the database.

The drastic changes between query and reference images of the Beirut database are shown in Figure 12. The first column in Figure 12 refers to the query images that were successfully recognized, and the 5 columns to the right of the green line represent the corresponding reference images used during the supervised learning. Note the visually apparent difficulty of the Beirut database with respect to difference in illumination, pose, and scale over ZuBuD. Despite all, the results achieved on Beirut database are quite acceptable.

## 5. Conclusion

We presented, in this paper, uSee, a supervised learning framework that selects in an energy aware fashion a reduced subset of relevant SIFT keypoints for image matching. With only 14.3% of an image SIFT keypoints and without any manual rotation for selected images as was done in prior



FIGURE 10: uSee failed images from the Beirut database.



FIGURE 11: Some ZuBuD query and reference images.

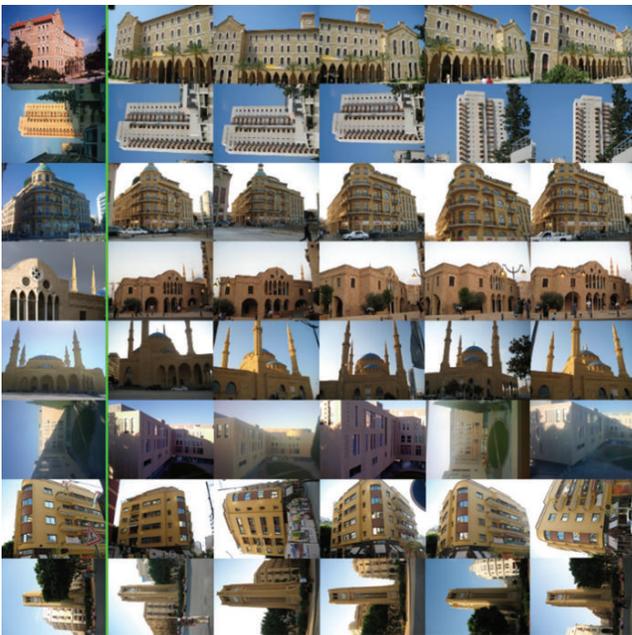


FIGURE 12: Some Beirut query and reference images.

research work, uSee exceeded published literature results and identified with an accuracy of 99.1% buildings in the Zurich Building database. Experimental results on Beirut Database showed that even with only 7% of SIFT keypoints, uSee can still deliver very good results in an energy aware paradigm,

TABLE 2: Effect of varying  $T_h$  on the recognition rate of the Beirut database.

Correlation threshold ( $T_h$ )	Recognition rate (%)
0.8	73.97%
0.85	79.45%
0.88	84.93%
0.89	78.08%
0.9	82.19%
0.91	78.08%
0.92	78.08%
0.95	73.97%

compared to the case where all SIFT keypoints are used for building recognition. With a more elaborate building segmentation in the preprocessing stage, uSee performance could be enhanced further more. Another extension of uSee that has been partially implemented is to redesign it and test it so that it completely runs on the mobile device, yet, the major future work planned for uSee is to architect it as an unsupervised learning framework and to compare it to feature saliency work.

## Acknowledgment

This work was supported by the Lebanese National Center for Scientific Research (LNCSR) grant to conduct and promote research at the American University of Beirut.

## References

- [1] L. Kennedy and M. Naaman, "Generating diverse and representative image search results for landmarks," in *Proceedings of the 17th International Conference on World Wide Web (WWW '08)*, pp. 297–306, April 2008.
- [2] T. Quack, B. Leibe, and L. V. Gool, "World scale mining of objects and events from community photo collections," in *Proceedings of the International Conference on Content-Based Image and Video Retrieval*, pp. 47–56, 2008.
- [3] Y.-T. Zheng, M. Zhao, Y. Song et al., "Tour the world: building a web-scale landmark recognition engine," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPR '09)*, pp. 1085–1092, June 2009.
- [4] G. Pass and R. Zabih, "Comparing images using joint histograms," *ACM Journal of Multimedia Systems*, vol. 7, no. 3, pp. 234–240, 1999.

- [5] H. Shao, T. Svoboda, T. Tuytelaars, and L. van Gool, "HPAT indexing for fast object/scene recognition based on local appearance," in *Proceedings of the 2nd International Conference on Image and Video Retrieval (CIVR '03)*, pp. 71–80, 2003.
- [6] H. Shao, T. Svoboda, and L. van Gool, "ZuBuD—Zurich buildings database for image based recognition," Tech. Rep. 260, Computer Vision Laboratory, Swiss Federal Institute of Technology, 2003, <http://www.vision.ee.ethz.ch/showroom/>.
- [7] W. Zhang and J. Kosecka, "Localization based on building recognition," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, p. 21, June 2005.
- [8] W. Zhang and J. Kosecka, "Experiments in building recognition," Tech. Rep. GMU-CS-TR-2004-3, 2004.
- [9] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.
- [10] J. J. Bonaiuto and L. Itti, "Combining attention and recognition for rapid scene analysis," in *Proceedings of the 3rd International Workshop Attention and Performance*, 2005.
- [11] K. Gao, S. Lin, Y. Zhang, S. Tang, and H. Ren, "Attention model based SIFT keypoints filtration for image retrieval," in *Proceedings of the 7th IEEE/ACIS International Conference on Computer and Information Science*, pp. 191–196, May 2008.
- [12] M. Awad and A. El Mobacher, "HISI: a soft framework for building recognition," in *Proceedings of the 5th International Conference on Soft Computing, Computing with Words and Perceptions in System Analysis, Decision and Control*, September 2009.
- [13] SIFT implementation in MATLAB by Andrea Vedaldi from the university of UCLA, <http://www.robots.ox.ac.uk/~vedaldi/code/sift.html>.

## Research Article

# Hyperspectral Image Classification Using Kernel Fukunaga-Koontz Transform

Semih Dinç<sup>1</sup> and Abdullah Bal<sup>2</sup>

<sup>1</sup> Department of Control and Automation Engineering, Yildiz Technical University, Esenler, 34220 Istanbul, Turkey

<sup>2</sup> Electronics and Communications Engineering, Yildiz Technical University, Esenler, 34220 Istanbul, Turkey

Correspondence should be addressed to Abdullah Bal; bal@yildiz.edu.tr

Received 6 May 2013; Accepted 18 September 2013

Academic Editor: Yudong Zhang

Copyright © 2013 S. Dinç and A. Bal. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper presents a novel approach for the hyperspectral imagery (HSI) classification problem, using Kernel Fukunaga-Koontz Transform (K-FKT). The Kernel based Fukunaga-Koontz Transform offers higher performance for classification problems due to its ability to solve nonlinear data distributions. K-FKT is realized in two stages: training and testing. In the training stage, unlike classical FKT, samples are relocated to the higher dimensional kernel space to obtain a transformation from non-linear distributed data to linear form. This provides a more efficient solution to hyperspectral data classification. The second stage, testing, is accomplished by employing the Fukunaga-Koontz Transformation operator to find out the classes of the real world hyperspectral images. In experiment section, the improved performance of HSI classification technique, K-FKT, has been tested comparing other methods such as the classical FKT and three types of support vector machines (SVMs).

## 1. Introduction

In the last decade, hyperspectral remote sensing technology has been included in popular study issues. Many articles have been proposed regarding hyperspectral images and spectral analysis since it offers new insight to various application areas such as agriculture [1], medical diagnose [2], illegal drug field detection [3], face recognition [4], and military target detection [5].

The idea behind the remote sensing technology relies on the relationship between photons and surface materials. Hyperspectral images are captured by the spectral sensors which are sensitive to larger portion of the electromagnetic spectrum than the traditional color cameras. While a digital color camera can capture only 3 bands (Red, Green, and Blue) in the range of 400 nm to 700 nm spectral wavelength, a typical hyperspectral sensor captures more than 200 bands within the range of 400 nm to 2500 nm. This means that HSI offers 200 or more features for an image pixel, instead of 3 values. HSI contains diverse information from a wide range of wavelengths. This characteristic yields more effective classification power for the application areas mentioned above.

Different type of materials can be represented by a set of bands which is called “spectral signature” that simplifies the separation of these materials.

There are also several challenges of the HSI to be solved. For instance, water absorption and some other environmental effects may induce some spectral bands to be noisy. These specific bands are called “noisy bands” and are required to be removed from the dataset. Another problem is the size of the data. Even in small scenes, hyperspectral images may have much larger size than traditional gray scale and color images, which means that the processing time is also longer than usual images. Detection of the redundant bands and removing them is crucial to reduce the number of features and total processing time. For this purpose, we refer to two papers [6, 7] to select best informative bands of our dataset.

In the literature, there are various classification techniques proposed for hyperspectral image classification problem, including neural networks, support vector machines, and Bayesian classifiers. In 2005, Benediktsson et al. [8] proposed a solution based on extended morphological models and neural networks. In 2007, Borges et al. [9] published their studies which is based on discriminative class learning

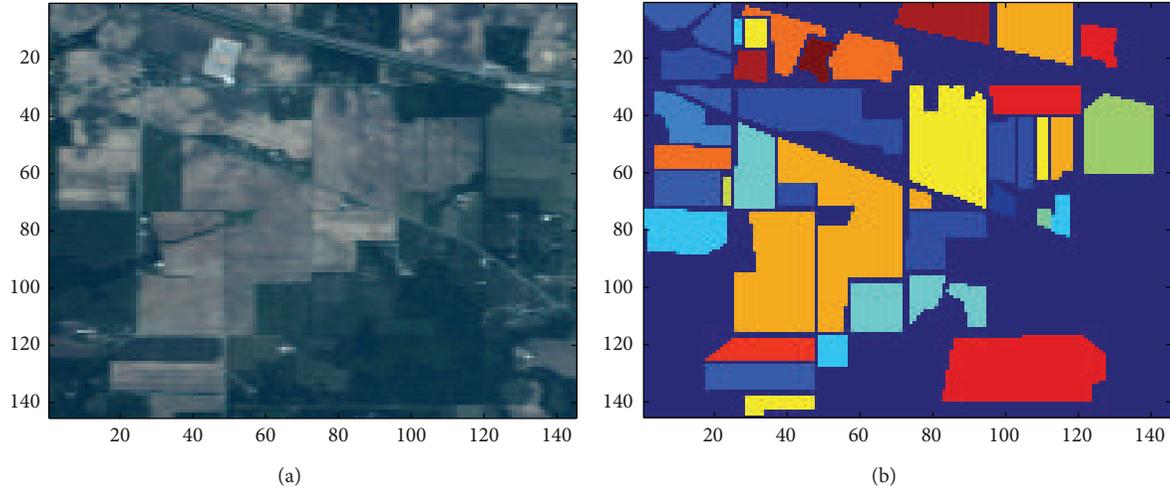


FIGURE 1: (a) RGB image of dataset, band numbers  $R = 24$ ,  $G = 14$ , and  $B = 8$ . (b) Ground truth data.

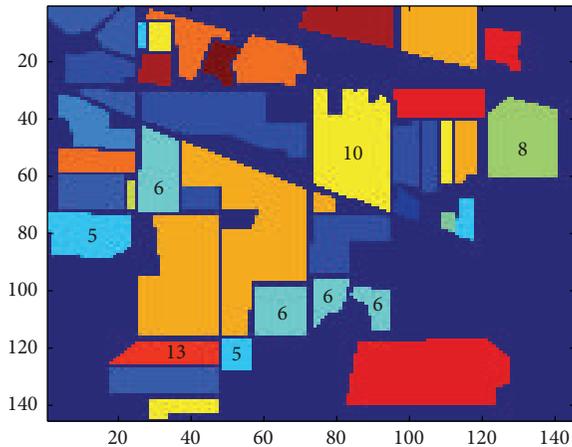


FIGURE 2: Ground truth data with selected classes.

using a new Bayesian based HSI segmentation method. In 2008, Alam et al. [10] proposed a Gaussian filter and post processing method for HSI target detection. Samiappan et al. [11] introduced a SVM based HSI classification approach which uses the same dataset used in this paper.

In this study, we present Kernel based Fukunaga-Koontz Transform method which is a novel solution to hyperspectral image classification. Classical FKT is a powerful method to solve two-pattern classification problems. However, when data is more complicated, classical FKT cannot produce satisfactory results. In this case, kernel transformations help FKT to increase separability of the data. In order to evaluate the K-FKT algorithm performance on HSI classifications, we select AVIRIS hyperspectral dataset which is a benchmark problem of this area. We have also used some other HSI dataset in our earlier studies and obtained high accuracy results which are presented in [12].

The remainder of this paper is organized as follows. The following section gives information about the contents of the AVIRIS dataset. A detailed description of the Kernel

Fukunaga-Koontz Transform is presented in Section 3 with training and testing stages. Classification results are given in Section 4, including the comparison with other methods. In the last section, we conclude our paper.

## 2. AVIRIS Dataset

This section includes detailed information about the AVIRIS Hyperspectral Image dataset which is called “Indian Pines” [13]. The dataset contains several different areas. Among these, there are mostly agricultural crop fields. Remaining parts have forests, highways, a rail line, and some low density housing. A convenient RGB colored view of the image can be seen in Figure 1(a).

Indian Pines dataset contains 16 different classes of crop fields. The ground truth data is shown in Figure 1(b). Table 1 shows the names of the classes and total number of samples for each class.

Basically, our dataset is  $145 \times 145 \times 220$  matrix that corresponds to 220 different bands of images having size of  $145 \times 145$ . In order to have more convenient form, this 3D matrix is transformed into 2D form as  $21025 \times 220$  matrix which indicates 21025 samples, and each sample has 220 numbers of features.

Before the classification processing, we removed regions that do not correspond to any class (dark blue areas in Figure 1(b)) from the dataset. Almost half of the samples do not belong to one of 16 classes. Once we eliminate these redundant bands, only 10336 samples are kept in the dataset.

## 3. Kernel Fukunaga-Koontz Transform

Traditional Fukunaga-Koontz Transform is a statistical transformation method which is a well-known approach [14–17], for two class classification problems. Basically, it operates by transforming data into a new subspace where both classes share the same eigenvalues and eigenvectors. While a subset of these eigenvalues can best represent ROI class, the

TABLE 1: Class names and number of samples.

Class number	Class	Samples
1	Alfalfa	54
2	Corn-notill	1434
3	Corn-mintill	834
4	Corn	234
5	Grass-pasture	497
6	Grass-trees	747
7	Grass-pasture-mowed	26
8	Hay-windrowed	489
9	Oats	20
10	Soybean-notill	968
11	Soybean-mintill	2468
12	Soybean-clean	614
13	Wheat	212
14	Woods	1294
15	Buildings-Grass-Trees-Drives	380
16	Stone-Steel-Towers	95
Total		10366

remaining eigenvalues represent the clutter class. With this characteristic, FKT differs from other methods.

Traditional FKT has been proposed to solve linear classification problems. When the data is nonlinearly distributed, the classical approach is not the best solution. Like other linear classifiers such as linear discriminant analysis (LDA), independent component analysis (ICA), and support vector machines (SVM), classical FKT suffers from nonlinearly distributed data. Therefore, in this paper, we used an improved version of classical FKT which basically changes the data distribution with a Kernel transformation to classify nonlinearly distributed data in a linear classification fashion. We will call it “K-FKT” in the rest of the paper. K-FKT algorithm consists of two stages: training and testing.

**3.1. Training Stage.** Since Fukunaga-Koontz Transform is a binary classification method, the training dataset is divided into two main classes. The region of interest (ROI) class is the first one to be classified. And the clutter class contains all other classes in the dataset except ROI. The algorithm is initiated by collecting an equal number ( $N$ ) of training samples for ROI and clutter classes that are represented as  $X$  and  $Y$ , respectively. Similarly,  $x_i$  and  $y_i$  are the training samples (or training signatures) of ROI and clutter classes as follows:

$$\begin{aligned} X &= [x_1, x_2, \dots, x_N], \\ Y &= [y_1, y_2, \dots, y_N]. \end{aligned} \quad (1)$$

The training sets  $X$  and  $Y$  are first normalized to avoid unexpected transformation results. Then they are mapped into higher dimensional Kernel space via the kernel transform. In simple terms, we assume that there is a mapping function  $\phi()$  to map all training samples to the Kernel space.

In this manner, we would obtain new training sets  $\tilde{X}$  and  $\tilde{Y}$  in which  $\phi(x_i)$  and  $\phi(y_i)$  denote the training samples in Kernel space. Equation (2) shows the mapping process. The symbol tilde “ $\sim$ ” indicates that corresponding variable is a kernel variable which has been transformed into the Kernel space as follows:

$$\begin{aligned} \tilde{X} &= [\phi(x_1), \phi(x_2), \dots, \phi(x_N)] = [\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_N], \\ \tilde{Y} &= [\phi(y_1), \phi(y_2), \dots, \phi(y_N)] = [\tilde{y}_1, \tilde{y}_2, \dots, \tilde{y}_N]. \end{aligned} \quad (2)$$

Unfortunately, such a mapping function is not available for many cases. Even if it was available, complexity of this operation would be very high since all training samples must be mapped to higher dimensional space separately. In order to overcome this problem, we may bypass the mapping function  $\phi()$  and get the same results in a faster way by using an approach called the “Kernel Trick” [16, 18]. According to the kernel trick, a kernel function is employed instead of mapping function. The following equation shows a generalized form of the kernel function:

$$K(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle, \quad (3)$$

where  $K$  is the kernel function with the parameters  $x_i$  and  $x_j$  which represent  $i$ th and  $j$ th training samples, respectively. In this paper we have examined two well-known kernel functions, Gaussian and Polynomial kernel. Gaussian kernel (4) relocates the samples in accordance to Gaussian distribution and employs “ $\sigma$ ” parameter to calibrate sensitivity as follows:

$$K(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right). \quad (4)$$

Polynomial Kernel is shown in (5). This function requires “ $d$ ” parameter to change the degree of the polynomial function and calibrate the sensitivity as follows:

$$K(x_i, x_j) = (x_i x_j + 1)^d. \quad (5)$$

In traditional FKT, computation of the covariance matrices of  $X$  and  $Y$  would be the next step. If we would apply the same operation to matrices  $\tilde{X}$  and  $\tilde{Y}$ , the results would be as follows:

$$\begin{aligned} T_0 &= \tilde{X}\tilde{X}^T, \\ C_0 &= \tilde{Y}\tilde{Y}^T, \end{aligned} \quad (6)$$

where  $T_0$  and  $C_0$  are the kernel matrices of the ROI and clutter classes, respectively. At this step, we are able to exploit the covariance properties to realize the Kernel Trick. As shown in (7), one of the kernel functions may be employed to complete the kernel transformation without requiring the mapping operation [19] as follows:

$$\begin{aligned} T_0(i, j) &= \tilde{x}_i \tilde{x}_j^T = K_T(i, j), \\ C_0(i, j) &= \tilde{y}_i \tilde{y}_j^T = K_C(i, j). \end{aligned} \quad (7)$$

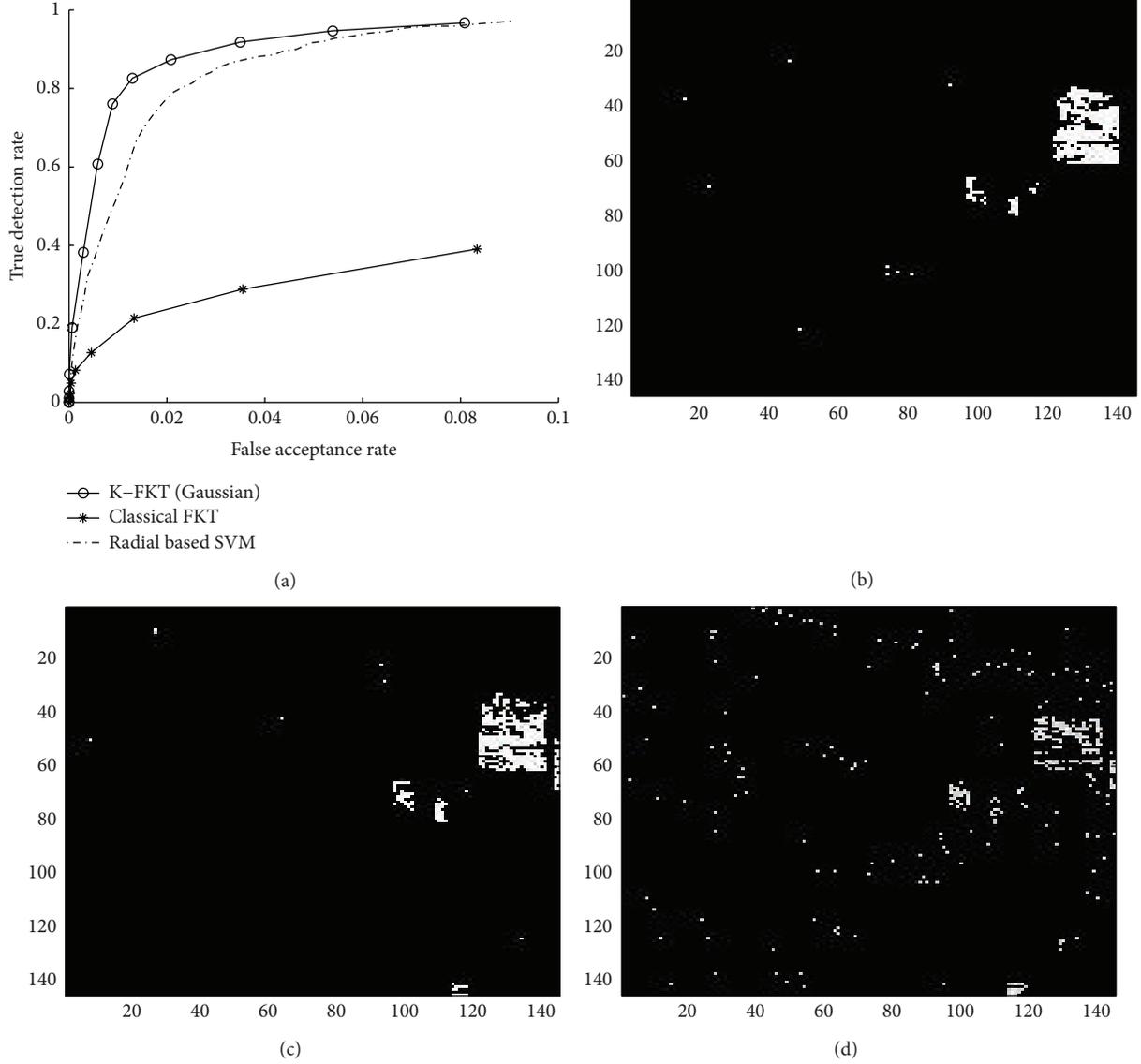


FIGURE 3: (a) Classification comparison for class number 8. (b), (c), and (d) Classification results of K-FKT, Classical FKT and Radial Based SVM, respectively.

After the kernel operations, the summation matrix of  $T_0$  and  $C_0$  is computed. Then it is decomposed into eigenvalues and eigenvectors as follows:

$$T_0 + C_0 = VDV^T, \quad (8)$$

where the symbols  $V$  and  $D$  represent eigenvector matrix and eigenvalue matrix, respectively. The diagonal elements of  $D$  are eigenvalues of the summation matrix. By using  $V$  and  $D$ , transformation operator  $P$  can be constructed as follows:

$$P = VD^{-1/2}. \quad (9)$$

After the multiplication by  $P$ , matrices  $T_0$  and  $C_0$  are transformed into eigenspace where both ROI and clutter

classes share the same eigenvalues and eigenvectors as follows:

$$\begin{aligned} T &= PT_0P^T, \\ C &= PC_0P^T, \end{aligned} \quad (10)$$

where  $T$  and  $C$  are the transformed matrices, respectively. Since they are transformed into the same eigenspace, the sum of matrices is equal to identity matrix as follows:

$$T + C = I. \quad (11)$$

Equation (11) implies that if  $\theta_i$  is an eigenvector of  $T$  and its corresponding eigenvalue is  $\lambda_i$ , then  $1 - \lambda_i$  is the eigenvalue

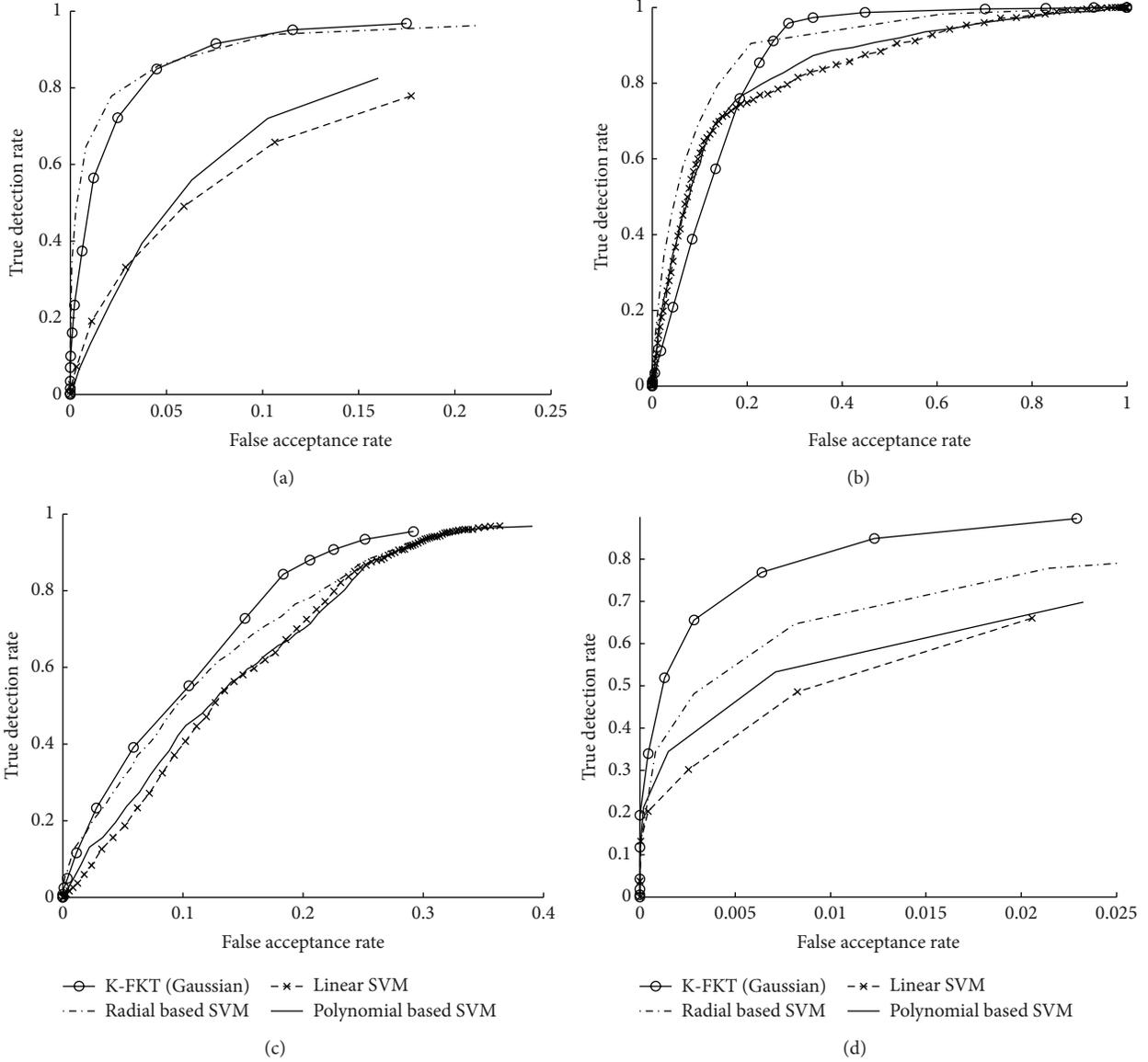


FIGURE 4: (a), (b), (c), and (d) Classification comparisons for classes number 5, 6, 10, and 13, respectively.

of  $C$  with the same eigenvector  $\theta_i$ . This relation can be represented as follows:

$$\begin{aligned}
 \lambda_i \theta_i &= T \theta_i, \\
 \lambda_i \theta_i &= (1 - C) \theta_i, \\
 (1 - \lambda_i) \theta_i &= C \theta_i.
 \end{aligned} \tag{12}$$

The above equations state that the more information an eigenvector contains about ROI class, the less information it has about clutter class. This characteristic evolves from the classical FKT algorithm.

**3.2. Testing Stage.** Testing stage starts with normalization of the test sample as it is done in the training stage. Similarly, the test sample must be mapped into Kernel space, but it is

not applicable due to the reasons explained in training stage. So we shall use kernel trick operation once more as follows:

$$Z = [K(x_1, z), K(x_2, z), \dots, K(x_N, z)]. \tag{13}$$

Equation (13) shows the kernel transformation of the test sample  $z$ . In the equation,  $x_i$  represents the ROI training samples and  $Z$  represents the kernel matrix of the corresponding test sample. Matrix  $Z$  is employed to calculate feature vector  $F$  in (16). Other factor required to calculate  $F$  is the normalized  $T_0$  matrix which is obtained by (14) as follows:

$$\hat{T} = T_0 - I_{1/N} T_0 - T_0 I_{1/N} + I_{1/N} T_0 I_{1/N}, \tag{14}$$

$$I_{1/N} = \frac{I_{N \times N}}{N}, \tag{15}$$

TABLE 2: Overall classification results.

Class	Precision	Recall	Accuracy
Alfalfa	1	1	1
Corn-notill	0.65	1	0.75
Corn-mintill	0.82	0.85	0.8
Corn	0.7	1	0.9
Grass-pasture	0.9	0.81	0.83
Grass-trees	1	0.85	0.84
Grass-pasture-mowed	1	1	1
Hay-windrowed	0.86	0.84	0.80
Oats	1	1	1
Soybean-notill	0.76	0.99	0.87
Soybean-mintill	0.79	0.94	0.82
Soybean-clean	0.8	1	0.77
Wheat	0.84	0.82	0.89
Woods	0.81	1	0.88
Buildings-Grass-Trees-Drives	0.84	0.95	0.88
Stone-Steel-Towers	1	0.94	0.91
Overall			0.86

where  $\hat{T}$  represents the normalized form. Once we have the matrices  $Z$  and  $\hat{T}$ , we are able to calculate the feature vector  $F$  as follows:

$$F_j = \frac{1}{\sqrt{\lambda_j}} \phi_j^T Z \quad j = 1, 2, \dots, N, \quad (16)$$

where  $\phi_j$  and  $\lambda_j$  denote the eigenvectors and eigenvalues of the  $\hat{T}$  matrix, respectively. The final step is the multiplication of the feature vector  $F$  by the transpose of eigenvector matrix of  $T$  as follows:

$$R = \Phi^T F, \quad (17)$$

where  $\Phi$  is the eigenvector matrix of  $T$  and  $R$  represents the result vector of test sample  $z$ . The test sample is estimated in ROI class if the  $L_2$  norm of  $R$  is a large value; otherwise it is estimated in clutter class. In order to summarize the proposed method and give a brief representation, steps of our algorithm are described below.

#### (1) Training stage

- (i) Select  $N$  number of training samples for ROI and Clutter class.
- (ii) Map training samples into Kernel space using “Kernel Trick” approach.
- (iii) Calculate the transformation  $P$  using eigenvalues and eigenvectors.
- (iv) Transform the the matrices  $T_0$  and  $C_0$  into the eigenspace via  $P$  operator.

#### (2) Testing stage

- (i) Map test sample  $z$  into Kernel space to obtain kernel matrix  $Z$ .
- (ii) Use  $Z$  and normalized ROI matrix  $\hat{T}$  to calculate feature vector  $F$ .
- (iii) Use  $F$  and eigenvalues of  $T$  to reach result value.
- (iv) Make the final decision by thresholding the result value.

## 4. Classification Results

In this section, classification results are presented. For each case, we selected a class among 16 classes and marked it as ROI class. Since it is not feasible to present the graphical results of all classes, we labeled some of the classes as in Figure 2. Particularly, we have shown the classified images of the class number 8 (Hay-windrowed). ROC curves are presented for the rest of labeled classes. Finally, we present Table 2 which includes the recall, precision, and accuracy results for all classes.

In first experiment “Hay-windrowed” class (labeled as number 8 in the figure) is selected as the ROI class. ROC curve results of three methods are shown in Figure 3(a). The results indicate that the kernel transformation remarkably improves the classification results. Also, our method shows higher accuracy than SVM at the same false acceptance rate (FAR) levels. For a better view of classification, result images are shown in Figures 3(b), 3(c), and 3(d) which are the results of K-FKT, Radial based SVM, and classical FKT, respectively. As shown in the figures, while classical FKT cannot classify the area, K-FKT and SVM classify the area with high accuracy.

Figures 4(a), 4(b), 4(c), and 4(d) show the ROC curves for other 4 classes, which are labeled as 5 (Grass-pasture), 6 (Grass-trees), 10 (Soybean-notill), and 13 (Wheat), respectively. According to the results, K-FKT presents higher accuracy than other classification methods. The results indicate that the ROC curve may vary for different classes since classes have different distributions.

Table 2 shows classification results of all classes. Precision, Recall, and Accuracy results are presented in each column. The results show that K-FKT offers promising classification capability for the hyperspectral image classification problem.

Experiments show that the overall accuracy of some specific classes such as “Corn-notill” and “Soybean-clean” is not higher than 80%. To clarify the ambiguity, we investigated the samples of these complicated classes and we realized that some classes are not in a “well-separable” condition. Our correlation analyses show that they are highly correlated with each other. It is the main reason behind the lower accuracy. In this study, we have only studied spectral features of the dataset, but employing also spatial features (e.g., neighbourhood information) may improve the results.

It is usually not an easy task to show a fair comparison of two studies due to unknown experimental parameters. However, by its experimental similarities, [11] can be considered as a comparison paper to our study. Samiappan et al. classify the

same dataset using SVMs with nonuniform feature selection. Their results present 75% of overall accuracy. Our results point out a remarkable contribution and exceed 75% accuracy by reaching 86% overall accuracy.

## 5. Conclusions

In this paper, we have presented a solution for hyperspectral image classification problems using supervised classification method called Kernel Fukunaga-Koontz Transform which is improved version of classical FKT. Since the classical FKT gives low performance for classification of nonlinearly distributed data, we have mapped the HSI data to higher dimensional Kernel space by kernel transformation. In that Kernel space, each region can be separated by classical FKT with higher performance. The experimental results verify that Kernel Fukunaga-Koontz Transform has higher classification performance than classical FKT, Linear, Polynomial, and Radial based SVM methods. Under these considerations, we can conclude that the Kernel FKT is a robust classification method for hyperspectral image classification. Our next goal is to use different kinds of kernel functions and investigate their effects to the classification results. We have an ongoing study that compares performances of different kernels.

## Acknowledgment

This research was supported by a Grant from The Scientific and Technological Research Council of Turkey (TUBITAK-112E207).

## References

- [1] P. S. Thenkabail, J. G. Lyon, and A. Huete, *Hyperspectral Remote Sensing of Vegetation*, CRC Press, New York, NY, USA, 2012.
- [2] L. Zhi, D. Zhang, J.-Q. Yan, Q.-L. Li, and Q.-L. Tang, "Classification of hyperspectral medical tongue images for tongue diagnosis," *Computerized Medical Imaging and Graphics*, vol. 31, no. 8, pp. 672–678, 2007.
- [3] M. Kalacska and M. Bouchard, "Using police seizure data and hyperspectral imagery to estimate the size of an outdoor cannabis industry," *Police Practice and Research*, vol. 12, no. 5, pp. 424–434, 2011.
- [4] Z. Pan, G. Healey, M. Prasad, and B. Tromberg, "Face recognition in hyperspectral images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 12, pp. 1552–1560, 2003.
- [5] D. Manolakis, D. Marden, and G. A. Shaw, "Hyperspectral image processing for automatic target detection applications," *Lincoln Laboratory Journal*, vol. 14, no. 1, pp. 79–116, 2003.
- [6] I. Faulconbridge, M. Pickering, and M. Ryan, "Unsupervised band removal leading to improved classification accuracy of hyperspectral images," in *Proceedings of the 29th Australasian Computer Science Conference (ACSC '06)*, V. Estivill-Castro and G. Dobbie, Eds., vol. 48 of *CRPIT*, pp. 43–48, Hobart, Australia, 2006.
- [7] O. Rajadell, P. Garca-Sevilla, and F. Pla, "Textural features for hyperspectral pixel classification," in *Pattern Recognition and Image Analysis*, H. Araujo, A. Mendona, A. Pinho, and M. Torres, Eds., vol. 5524 of *Lecture Notes in Computer Science*, pp. 208–216, Springer, Berlin, Germany, 2009.
- [8] J. A. Benediktsson, J. A. Palmason, and J. R. Sveinsson, "Classification of hyperspectral data from urban areas based on extended morphological profiles," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 43, no. 3, pp. 480–491, 2005.
- [9] J. Borges, J. Bioucas-Dias, and A. Maral, "Bayesian hyperspectral image segmentation with discriminative class learning," in *Pattern Recognition and Image Analysis*, J. Mart, J. Bened, A. Mendona, and J. Serrat, Eds., vol. 4477 of *Lecture Notes in Computer Science*, pp. 22–29, Springer, Berlin, Germany, 2007.
- [10] M. S. Alam, M. N. Islam, A. Bal, and M. A. Karim, "Hyperspectral target detection using Gaussian filter and post-processing," *Optics and Lasers in Engineering*, vol. 46, no. 11, pp. 817–822, 2008.
- [11] S. Samiappan, S. Prasad, and L. M. Bruce, "Automated hyperspectral imagery analysis via support vector machines based multi-classifier system with non-uniform random feature selection," in *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS '11)*, pp. 3915–3918, July 2011.
- [12] S. Dinc, *Hyperspectral target recognition using multiclass kernel fukunagakoontztransform [M.S. thesis]*, Graduate School of Natural and Applied sciences, Yildiz Technical University, İstanbul, Turkey, 2011.
- [13] Aviris hyperspectral data, 2013, <http://aviris.jpl.nasa.gov/html/data.html>.
- [14] K. Fukunaga and W. L. G. Koontz, "Application of the Karhunen-Loève expansion to feature selection and ordering," *IEEE Transactions on Computers*, vol. 19, no. 4, pp. 311–318, 1970.
- [15] S. Ochilov, M. S. Alam, and A. Bal, "Fukunaga-koontz transform based dimensionality reduction for hyperspectral imagery," in *Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery XII*, vol. 6233 of *Proceedings of SPIE*, April 2006.
- [16] Y.-H. Li and M. Savvides, "Kernel fukunaga-koontz transform subspaces for enhanced face recognition," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '07)*, pp. 1–8, June 2007.
- [17] W. Zheng and Z. Lin, "A new discriminant subspace analysis approach for multi-class problems," *Pattern Recognition*, vol. 45, no. 4, pp. 1426–1435, 2012.
- [18] R. Liu and H. Zhi, "Infrared point target detection with fisher linear discriminant and kernel fisher linear discriminant," *Journal of Infrared, Millimeter, and Terahertz Waves*, vol. 31, no. 12, pp. 1491–1502, 2010.
- [19] D. Tuia and G. Camps-Valls, "Urban image classification with semisupervised multiscale cluster kernels," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 4, no. 1, pp. 65–74, 2011.

## Research Article

# Solving a Novel Inventory Location Model with Stochastic Constraints and $(R, s, S)$ Inventory Control Policy

**Guillermo Cabrera,<sup>1,2</sup> Pablo A. Miranda,<sup>2</sup> Enrique Cabrera,<sup>2,3</sup> Ricardo Soto,<sup>2,4</sup> Broderick Crawford,<sup>2,5</sup> Jose Miguel Rubio,<sup>2,6</sup> and Fernando Paredes<sup>7</sup>**

<sup>1</sup> Department of Engineering Science, University of Auckland, Auckland 1010, New Zealand

<sup>2</sup> Pontificia Universidad Católica de Valparaíso, Valparaíso 2362807, Chile

<sup>3</sup> CIMFAV, Facultad de Ingeniería, Universidad de Valparaíso, Valparaíso 2340000, Chile

<sup>4</sup> Universidad Autónoma de Chile, Santiago 7500000, Chile

<sup>5</sup> Universidad Finis Terrae, Santiago 7500000, Chile

<sup>6</sup> Universidad de Playa Ancha, Valparaíso 33449, Chile

<sup>7</sup> Escuela de Ingeniería Industrial, Universidad Diego Portales, Santiago 8370179, Chile

Correspondence should be addressed to Guillermo Cabrera; [guillermo.cabrera@ucv.cl](mailto:guillermo.cabrera@ucv.cl)

Received 3 May 2013; Revised 9 August 2013; Accepted 9 August 2013

Academic Editor: Vishal Bhatnaga

Copyright © 2013 Guillermo Cabrera et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We solve a novel inventory-location model with a stochastic capacity constraint based on a periodic inventory control (ILM-PR) policy. The ILM-PR policy implies several changes with regard to other previous models proposed in the literature, which consider continuous review as their inventory policy. One of these changes is the inclusion of the undershoot concept, which has not been considered in previous ILM models in the literature. Based on our model, we are able to design a distribution network for a two-level supply chain, addressing both warehouse location and customer assignment decisions, whilst taking into consideration several aspects of inventory planning, in particular, evaluating the impact of the inventory control review period on the network configuration and system costs. Because the model is a very hard-to-solve combinatorial nonlinear optimisation problem, we implemented two heuristics to solve it, namely, Tabu Search and Particle Swarm Optimisation. These approaches were tested over small instances in which they were able to find the optimal solution in just a few seconds. Because the model is a new one, a set of medium-size instances is provided that can be useful as a benchmark in future research. The heuristics showed a good convergence rate when applied to those instances. The results confirm that decision making over the inventory control policy has effects on the distribution network design.

## 1. Introduction

Distribution network design (DND) is one of the most important problems for companies that distribute products to their customers. The problem consists of selecting specific sites to install plants, warehouses, and distribution centres, assigning customers to serving and interconnecting facilities by flow assignment decisions. DND is typically solved as part of a sequential approach that simplifies associated tactical and operational issues. Hence, the decisions that have been omitted are tackled only after the DND problem has already been solved. This means that decisions related to the location of warehouses and allocation of customers are made without

taking into account operational aspects such as transportation and inventory. This situation leads to suboptimal designs because operational decisions are restricted to the current network design.

This paper considers a two-level supply chain, in which a single plant serves a set of warehouses, which in turn serve a set of end customers or retailers. Unlike traditional approaches and in accordance with the recent inventory-location literature, in this paper, we incorporate the inventory control policy as a relevant factor that directly affects DND. A distinctive feature of our model is that it is based on a periodic inventory control policy  $(R, s, S)$  for each distribution centre (DC) in a single-product scenario, which is important for

those industries in which a continuous revision policy is not possible. Because our model is a very hard-to-solve nonlinear one, we have solved it using well-known heuristic approaches, namely, Tabu Search (TS) and Particle Swarm Optimisation (PSO). Moreover, as the model was recently developed by us, there is no set of instances that can be used as a benchmark. Thus, in this paper, we present two types of medium-sized instances. The first type considers uniformly distributed customers. The other one considers two clusters of customers. Warehouses are uniformly distributed in both types of instances.

The remainder of the paper is organised as follows. First, a literature review of the main ILM is presented in Section 2. Then, in Section 3, we present and analyse the stochastic capacity constraint under periodic review  $(R, s, S)$ . At the end of that section, the formulation of the model is presented. Section 4 briefly presents a review of both TS and PSO algorithms and some features of the corresponding implementations. Section 5 starts explaining the procedure used to generate the set of instances presented in this paper. In Section 5.1, the obtained results are presented. Finally, in the last section, some conclusions based on the numerical results are outlined.

## 2. Literature Review

Many authors have been focused on the DND problem and on ILM in particular during the last 15 years. For instance Simchi-Levi and Zhao [1], Mourits and Evers [2], Bradley and Arntzen [3], Miranda and Garrido [4], and Miranda [5] all analysed the levels of decision making related to DND and supply chain management (SCM). Daskin [6], Simchi-Levi et al. [7], and Drezner and Hamacher [8] present detailed FLP reviews and analysis. However, the traditional structure of FLP is not useful for considering interactions between facility location and inventory decisions or the impact of the latter on network configurations. For example, the risk pooling effect states that the total system safety stock is reduced when customers are served by a smaller number of warehouses. Daskin et al. [9] and Shen et al. [10] incorporate an  $(Q, RP)$  inventory control policy into the widely studied uncapacitated facility location problem (UFLP), establishing a safety stock at each site. Daskin et al. [9] employ Lagrangian relaxation to solve the model, whereas Shen et al. [10] reformulate the model as a set-covering problem and solve it using a column generation method. Based on the same inventory control policy, Miranda and Garrido [4] consider the order quantity for each warehouse as a decision variable and the capacitated facility location problem (CFLP) as a base framework. Finally, in Miranda and Garrido [11] and Ozsen et al. [12] authors handle capacity constraints by using previous inventory-location models.

More recently, Kumar and Tiwari [13] have presented an ILM that incorporates the risk pooling effect for both safety stock and running inventory. Additionally, in their model, the authors consider the effect produced when warehouses and end customers work jointly. Tancrez et al. [14] have presented a three-level supply chain nonlinear ILM that integrates location, allocation, and shipment sizes. In [15], the authors developed a model for the DND problem that considers the

short lifetime of perishable products. To solve their model, the authors implemented a Lagrangian relaxation. One paper that has addressed the DND problem using periodic 3 inventory review is presented in [16]. In their paper, the authors consider a  $(R, S)$  inventory policy, which is slightly different from the one that is considered in this work.

The authors have presented different techniques to solve these models. For instance, Bard and Nananukul [17] proposed a branch and price algorithm for an integrated production and inventory routing problem. In [18], Lagrangian relaxation was used to solve a mixed integer linear programming model for multiple echelon and multiple commodity supply chain network design. Heuristics have also been used to solve DND problems. For instance, Armentano et al. [19] TS is used to minimise the production and inventory cost in a model that integrates production and distribution decisions by considering the capacity constraints of the plant. Askin et al. [20] implemented an evolutionary algorithm to solve an ILM considering multicommodity and distribution planning decisions. In their paper, the authors present a very comprehensive description of their genetic algorithm. To the best of our knowledge, PSO has not been considered to solve ILM problems.

As we stated previously, in this paper, we have considered a two-level supply chain with a single plant, a set of warehouses, and a set of end customers or retailers. Because we incorporate the inventory control policy as a relevant factor that directly affects DND in this paper, based on a periodic inventory control policy  $(R, s, S)$  for each Distribution Center (DC), this model can be considered as a variant of the models presented previously in the literature [4, 9, 12, 21], which considers a policy of continuous inventory review  $(Q, R)$ . In next section, the new model proposed in this work is presented.

## 3. Inventory-Location Model with Stochastic Capacity Constraint

The model presented in [11] optimises warehouse location and customer assignment decisions, taking into account fixed installation, transportation, inventory, and fixed ordering costs. The authors assume that each warehouse  $i$  operates a continuous inventory review policy based on a  $(Q, RP)$  policy to meet a stochastic demand, with mean  $D$  (units of product per time unit) and variance  $V$ . It is also assumed that the plant takes a lead time,  $LT$ , to fill incoming orders from warehouse  $i$ . Stochastic constraint on inventory capacity is proposed, assuming a maximum inventory level for each warehouse  $I^{\text{cap}}$ . This constraint is based on chance constrained programming [22] and it fixes a maximum probability,  $\beta$ , of violating inventory capacity at peak times, which occurs only when orders arrive at the warehouses. This inventory level corresponds to the reorder point  $RP$ , which is stated in order to satisfy demand during  $LT$  with at least a probability of  $1 - \alpha$ , minus stochastic demand during  $LT$ , plus order quantity,  $RP - SD(LT) + Q$ . Thus, the inventory capacity constraint can be written as a deterministic nonlinear constraint as follows:

$$Q + (Z_{1-\alpha} + Z_{1-\beta}) \cdot \sqrt{LT} \cdot \sqrt{V} \leq I^{\text{cap}}. \quad (1)$$

It may be noted that a similar and more conservative capacity constraint is proposed in [12], which ensures that inventory capacity is observed in 100% of the cases. However, when periodic review is considered, particularly assuming a  $(R, s, S)$  inventory control policy [23], capacity constraint cannot be stated at any moment and does not take the same form as in (1). In a  $(R, s, S)$  inventory control policy, inventory levels are reviewed only every  $R$  period, and if the inventory level is lower than  $s$ , then an order is submitted to reach the objective level  $S$ . Consequently, order size must consider the well-known undershoot magnitude (US), which is the amount of items required in addition to  $s$  to reach  $S$  units of inventory. The average US as a function of demand mean and variance,  $D$  and  $V$ , and for a review period  $R$ , can be computed as

$$US(D, V) = \frac{V}{2 \cdot D} + \frac{R \cdot D}{2}. \quad (2)$$

In terms of inventory capacity constraints, peak inventory levels are not controlled at any time, but only at specific times for each review period. The peak inventory level is reached only when orders arrive at the warehouse  $LT$  time units after the last order and naturally only if an order was submitted to the central warehouse or plant. Consequently, when an order arrives at a warehouse, the inventory level is

$$\begin{aligned} & \underbrace{(s - US)}_{\text{Inventory level when an order is submitted}} \\ & + \underbrace{\frac{(S - s + US)}{\text{Submitted order size}}}_{\text{Submitted order size}} - \underbrace{\frac{SD(LT)}{\text{Stochastic demand during LT}}}_{\text{Stochastic demand during LT}} \quad (3) \\ & = \frac{\underbrace{S - SD(LT)}_{\text{Maximum inventory level when an order arrives}}}{\text{Maximum inventory level when an order arrives}}. \end{aligned}$$

This expression is not surprising, as when an order is submitted to the plant, it is necessary that the total inventory position (on hand plus on order inventory) reaches level  $S$  and that  $LT$  time units later, the inventory level is reduced by the demand during the  $LT$ ,  $SD(LT)$ . Similar to [11], in this paper, we propose that this constraint must be observed for each peak inventory instant (i.e., for each order period) with a fixed and known probability of  $1 - \beta$ , but now assuming a periodic review as follows:

$$S \leq I^{\text{cap}} + D \cdot LT - Z_{1-\beta} \cdot \sqrt{V \cdot LT}. \quad (4)$$

We then define the minimum order size,  $Q$ , as

$$S = s + Q. \quad (5)$$

Consequently, constraint (4) can be rewritten as follows:

$$Q + s \leq I^{\text{cap}} + D \cdot LT - Z_{1-\beta} \cdot \sqrt{V \cdot LT}. \quad (6)$$

Finally, the reorder point  $s$  is set to ensure that, for each time an order is not submitted (inventory level larger than  $s$ ), the inventory level is large enough to fill the demand until the next order arrives in  $R + LT$  time units, with a probability or service level of  $1 - \alpha$ :

$$s = D \cdot (LT + R) + Z_{1-\alpha} \cdot \sqrt{R + LT} \cdot \sqrt{V}. \quad (7)$$

Finally, substituting (7) into (6), the inventory capacity constraint is

$$\begin{aligned} & Q + D \cdot R + \left( Z_{1-\alpha} \cdot \sqrt{R + LT} + Z_{1-\beta} \cdot \sqrt{LT} \right) \\ & \cdot \sqrt{V} \leq I^{\text{cap}}. \end{aligned} \quad (8)$$

According to the previous inventory control assumption, we describe the proposed Inventory-Location Model with Stochastic Constraints on Inventory Capacity under Periodic Review (ILM-SCC-PR) as a stochastic nonlinear mixed-integer programming model (SNL-MIP). Based on a periodic  $(R, s, S)$  inventory control policy, the safety stock to be included in the objective function is the average inventory level just before an order arrives at the warehouse:

$$\begin{aligned} & (s - US) - D \cdot LT \\ & = D \cdot R + Z_{1-\alpha} \cdot \sqrt{R + LT} \cdot \sqrt{V} - \left( \frac{V}{2 \cdot D} + \frac{D \cdot R}{2} \right). \end{aligned} \quad (9)$$

Additionally, inventory and ordering costs related to order size or inventory cycle are evaluated in terms of the minimum order size  $Q$  as a decision variable, as in the well-known EOQ model:

$$\frac{OC \cdot D}{(Q + US)} + \frac{HC \cdot (Q + US)}{2}. \quad (10)$$

The variables and parameters considered in the mathematical formulation are as follows.

#### Parameters

- $N$ : Number of available sites to install warehouses
- $M$ : Number of customers that must be served by the installed warehouses
- $F_i$ : Fixed location cost of a warehouse on site  $i$  (\$/day)
- $C_{ij}$ : Transportation unit cost from the warehouse on site  $i$  to customer  $j$  (\$/unit)
- $OC_i$ : Fixed ordering cost at site  $i$  (\$/order)
- $HC_i$ : Holding cost per time unit at site  $i$  (\$/day)
- $\mu_j$ : Mean demand of customer  $j$  per day
- $\sigma_j$ : Standard deviation of the demand of customer  $j$  per day
- $I_i^{\text{cap}}$ : Capacity at warehouse  $i$
- $Q_i^{\text{cap}}$ : Maximum order size at warehouse  $i$
- $R_i$ : Inventory check period at warehouse  $i$  in days
- $LT_i$ : Average leadtime at warehouse  $i$  in days
- $Z_{1-\alpha}$ : Value of the Standard Normal Distribution, which accumulates a probability of  $1 - \alpha$
- $Z_{1-\beta}$ : Value of the Standard Normal Distribution, which accumulates a probability of  $1 - \beta$ .

### Variables

- $X_i$ : Binary variable. It is equal to 1 if a warehouse is installed on site  $i$  and 0 otherwise
- $Y_{ij}$ : Binary variable. It is equal to 1 if warehouse  $i$  serves customer  $j$  and 0 otherwise
- $Q_i$ : Order size at warehouse  $i$ . It is greater than 0 if  $D_i > 0$ , and 0 otherwise
- $D_i$ : Total mean demand at warehouse  $i$ . It is greater than 0 if there exist at least one  $Y_{ij} > 0$  and 0 otherwise
- $V_i$ : Total variance of the demand at warehouse  $i$ . It is greater than 0 if  $D_i > 0$  and 0 otherwise.

Then, the optimisation problem is

Min

$$\begin{aligned} & \sum_{i=1}^N (F_i \cdot X_i) + \sum_{i=1}^N \sum_{j=1}^M (C_{ij} \cdot Y_{ij}) \\ & + \sum_{i=1}^N \left( \frac{OC_i \cdot D_i}{(Q_i + US_i)} + \frac{HC_i \cdot (Q_i + US_i)}{2} \right) \\ & + \sum_{i=1}^N \left( HC_i \left( D_i \cdot R_i + Z_{1-\alpha} \cdot \sqrt{R_i + LT_i} \cdot \sqrt{V_i} - US_i \right) \right) \end{aligned} \quad (11)$$

s.t.:

$$\sum_{i=1}^N Y_{ij} = 1 \quad \forall j = 1, 2, \dots, M, \quad (12)$$

$$Y_{ij} \leq X_i \quad \forall i = 1, 2, \dots, N, \quad \forall j = 1, 2, \dots, M, \quad (13)$$

$$Q_i + D_i \cdot R_i + \left( Z_{1-\alpha} \cdot \sqrt{R_i + LT_i} + Z_{1-\beta} \cdot \sqrt{LT_i} \right) \cdot \sqrt{V_i} \leq I_i^{\text{cap}} \cdot X_i \quad \forall i = 1, 2, \dots, N, \quad (14)$$

$$Q_i + US_i \leq Q_i^{\text{cap}} \cdot X_i \quad \forall i = 1, 2, \dots, N, \quad (15)$$

$$D_i = \sum_{j=1}^M \mu \cdot Y_{ij} \quad \forall i = 1, 2, \dots, N, \quad (16)$$

$$V_i = \sum_{j=1}^M \sigma_j^2 \cdot Y_{ij} \quad \forall i = 1, 2, \dots, N, \quad (17)$$

$$US_i = \begin{cases} \frac{V_i}{2 \cdot D_i} + \frac{R_i \cdot D_i}{2} & \text{if } D_i > 0 \\ 0 & \text{otherwise} \end{cases} \quad (18)$$

$$X_i, Y_{ij} \in \{0, 1\} \quad \forall i = 1, \dots, N; \quad \forall j = 1, \dots, M. \quad (19)$$

Equation (11) is the total system cost. The first term is the fixed setup and operating cost when opening warehouses. The second term is the daily transport cost between warehouse and customers. The third term contains fixed order and

inventory costs related to warehouse order size. The fourth term represents the storage cost associated with safety stock at each warehouse. Equation (12) ensures that retailers are served by only one warehouse. Inequality (13) ensures that customers are only assigned to installed warehouses ( $X_i = 1$ ). Inequality (14) ensures that the inventory capacity of each warehouse is respected at least with a probability  $1 - \alpha$ . Inequality (15) ensures that the order size is below the maximum order size  $Q_i^{\text{cap}}$  allowed to warehouse  $i$ . Equations (16) and (17) compute the mean and variance of the total demand served by each warehouse  $i$ , respectively. Equations (18) determine the value of  $US_i$ . Finally, (19) states integrality ( $0 - 1$ ) for the variables  $Y_{ij}$  and  $X_i$ . This model is NP hard because it is clearly an extension of the UFLP, which is already NP hard (UFLP can be obtained just by using  $HC = 0$  and  $OC = 0$ ). In addition, the objective function and one constraint are nonlinear, resulting in a model that is very difficult to solve to optimality using classical mathematical programming techniques. Therefore, in this paper, we attempt to solve medium-size instances by mean of two well-known heuristics called Tabu Search (TS) and Particle Swarm Optimisation (PSO).

## 4. Heuristic Methods

Heuristic methods are a common approach to solve hard combinatorial optimisation problems. Despite the fact that heuristics do not guarantee optimality, the solutions provided by them can be considered as good suboptimal ones. In contrast exact methods guarantee optimality; however, they usually fail when dealing with medium- and large-sized problems. In this paper, two heuristics have been separately considered to solve our DND problem. The first one corresponds to a well-known local search called Tabu Search. The second one is an evolutionary algorithm called Particle Swarm Optimisation. In the next subsections, an overview of these two heuristics is provided.

**4.1. Tabu Search.** We can describe the TS approach as a local search technique guided by the use of adaptive or flexible memory structures. However, such a general definition fails to show the specificity of TS and could be confused with other types of Greedy Random Adaptive Search Procedure (GRASP) algorithms. The variety of the tools and search principles introduced and described in [24] are such that the TS can be considered as the seed of a general framework for modern heuristic search [25]. TS has been applied to several combinatorial optimisation problems (see, for instance, [26–29]). So too other techniques [30–32] have been used to solve different variants of the FLP. The TS is essentially a local search algorithm; that is, it needs to “exchange information” with its neighbours. To do that, first, the neighbourhood must be defined. Typically, TS algorithm has only one neighbourhood move that defines a set of possible candidate solutions. This move is used across all executions of the TS algorithm. Because different neighbourhood definitions can lead to different results in this paper, we have implemented two types of neighbourhoods. The first one is defined by a change in the allocation of a customer from one facility to another without

any restriction; that is, the new facility could be either open or closed at the allocation moment. The second one is defined by a change in the allocation of a customer from one facility to another in which the new facility was opened previous to the movement 90% of the time; in other words, the probability of moving a customer to a closed facility is equal to 10%. The TS heuristic provides a diversification mechanism that allows it to get out of low-quality neighbourhoods and “jump” to explore new neighbourhoods. The diversification mechanism implemented in this study is a restart method, which re-initialises a current solution without losing the best solution found by the algorithm. As stated above, the restart method is used in case the TS is unable to move out of low-quality neighbourhoods. The TS algorithm requires the following structures to be implemented.

- (i) *solutionVector*: a vector with a size equal to the number of customers. For each customer this vector contains the DC to which the customer is allocated.
- (ii) *bestSolutionVector*: a vector corresponding to the best *solutionVector* found during the TS execution.
- (iii) *candidateSolutionVector*: a vector resulting from application of the neighbourhood movement to a *solutionVector*.
- (iv) *candidateList*: a list of *candidateSolutionVectors*. From this list, we obtain the next *solutionVector*, which corresponds to the best *candidateSolution* from the list.
- (v) *tabuList*: an  $N \times M$  matrix that contains those neighbourhood movements that are prohibited in the current TS iteration.

Additionally, our TS implementation also requires the following parameter list.

- (i) *iterationNumber*: Total iteration number to end the algorithm.
- (ii) *diversificationBound*: total iteration number to apply diversification criterion (restart method).
- (iii) *tabuListSize*: size of *tabuList*. This means the number of iterations for which a specific movement remains banned.
- (iv) *listOfCandidates*: size of the set of *candidateList*.

The TS algorithm starts with a random solution in which each customer is allocated to a randomly selected DC. When all customers are assigned, TS validates the model constraints. If the constraints are satisfied, the initial solution ( $S_0$ ) is assigned as a best solution ( $S_{\text{best}}$ ). A list of candidate solutions ( $NS_0$ ) is generated after that. They are neighbours of  $S_0$ . The size of the  $NS_0$  set corresponds to the *listOfCandidates* parameter. Once the set of  $NS_0$  has been generated, the TS algorithm selects the best candidate solution ( $NS_0^{\text{best}}$ ), which should not stay on the tabu list. If the  $NS_0^{\text{best}}$  actually is in the *tabuList*, it cannot be selected unless its cost is less than the cost of the  $S_{\text{best}}$  found until this moment (*aspiration criterion*). When a new current solution  $S_1$  is selected, the *tabuList*

is updated, and a new iteration starts. The algorithm stops when the number of iterations  $k$  is equal to the parameter *iterationNumber*. Below, we can see the general framework of our TS implementation.

**4.2. Particle Swarm Optimisation.** The PSO algorithm was proposed and developed by Kennedy and Eberhart [33–37]. Although the PSO algorithm was initially designed to tackle continuous optimisation problems, during the last decade, it has proven to be a very good alternative for solving combinatorial optimisation problems. In fact, several articles have used this technique to solve complex combinatorial optimisation problems; see, for example, [38, 39]. For a comprehensive analysis of publications concerning PSO approaches, see [40].

Unlike other evolutionary algorithms, PSO does not use the “survival” concept. This is because all particles are kept “alive” throughout the algorithm execution time and at no time is their survival threatened. The algorithm starts when a set of particles  $\mathcal{S}$  (or *swarm*) is initialised. Each particle  $p^k \in \mathcal{S}$ , with  $k = 1, \dots, P$ , is represented by a vector in which the value of each element  $p_j^k$  with  $j = 1, \dots, M$  corresponds to the warehouse that has been set to a customer  $j$ . In our algorithm, the initialisation is a random process. For each particle, we know its current position (denoted by  $p^k$ ) and its previous best position (denoted by  $p^{k(\text{best})}$ ). We also calculate for each particle the best current neighbour (denoted by  $p^{k(\text{neighbour})}$ ) and the best particle (denoted by  $p^{\text{best}}$ ) found so far. The position of each individual particle is adjusted at each iteration by considering its previous best positions, the position of the best neighbour, and the position of the global best solution found so far (see Algorithm 1).

The general frame for our PSO algorithm is presented below.

The most important step in Algorithm 2 is the particle updating. As we have said before, the PSO algorithm was designed for continuous optimisation problems. Therefore, we need to adapt some of its steps to solve combinatorial problems. One important change is in the strategy used to move particle  $p^k$  from its position in the iteration  $t$  to its next position in iteration  $t+1$ . To do that, we have considered three possible alternatives: the first one (namely  $s_1$ ) corresponds to a change in the allocation of one of the customers (randomly selected) to a new (randomly selected) warehouse. The second alternative ( $s_2$ ) is a change in the allocation of one of the customers (randomly selected) to a new warehouse following the allocation of either the bestNeighbour ( $p^{k(\text{neighbour})}$ ) or the prior best position ( $p^{k(\text{best})}$ ). Finally, the third alternative ( $s_3$ ) corresponds to the same movement but now considering the allocation from the global best solution reached so far  $p^{\text{best}}$ . Therefore, the position of particle  $p^k$  in the iteration  $t+1$  will be the best particle among  $s_1$ ,  $s_2$ , and  $s_3$ . Below we present the pseudocode for our particle updating process (see Algorithm 3).

As with local search algorithms, the PSO approach can also get trapped in local optima if the global best and local best positions are equal to the position of the particle over

```

begin
   $k = 0$ ;
  Generate initialSolution  $S_k$ ;
   $S_{\text{best}} = S_k$ ;
  while  $k < \text{iterationNumber}$  do
     $NS_k = \text{Generate Neighbourhood}(S_k)$ ;
    find best ( $NS_k$ );
    while  $NS_k^{\text{best}}$  isTabu do
      if cost of  $NS_k^{\text{best}} < \text{cost of } S_{\text{best}}$  then
        aspirationCriterion;
        replace  $NS_k^{\text{best}}$ ;
      If cost of  $ns_k^{\text{best}} < \text{cost of } S_{\text{best}}$  then
         $S_{\text{best}} = ns_k^{\text{best}}$ ;
        iteration without improvements = 0;
        iteration without improvements + 1;
       $S_k = ns_k^{\text{best}}$ ;
      update (tabuList);
      check (diversificationCriterion);
       $k = k + 1$ 
  end

```

ALGORITHM 1: General algorithmic frame for tabu search.

```

begin
  Initialise set  $S$ ;
  Calculate  $p^{\text{best}}$ ;
  while maxIteration not reached do
    foreach  $p^k \in S$  do
      Calculate  $p^{k(\text{best})}$ ;
      Calculate  $p^{k(\text{neighbour})}$ ;
      Update  $p^k$ ;
      Calculate  $p^{\text{best}}$ ;
  end

```

ALGORITHM 2: General algorithmic frame for particle swarm optimisation.

```

begin
  Particles  $s_1, s_2, s_3$ ;
  If inertia > rand[0, 1] then
     $s_1 = \text{LocalMove}(p^k)$ ;
  If cognitiveFactor > rand[0, 1] then
     $s_2 = \text{LocalMove}(p^{k(\text{best})})$ ;
   $s_2 = \text{LocalMove}(p^{k(\text{neighbour})})$ ;
  If globalCognitiveFactor > rand[0, 1] then
     $s_3 = \text{LocalMove}(p^{\text{best}})$ ;
   $p^k = \text{best}(s_1, s_2, s_3)$ ;
end

```

ALGORITHM 3: Particle update.

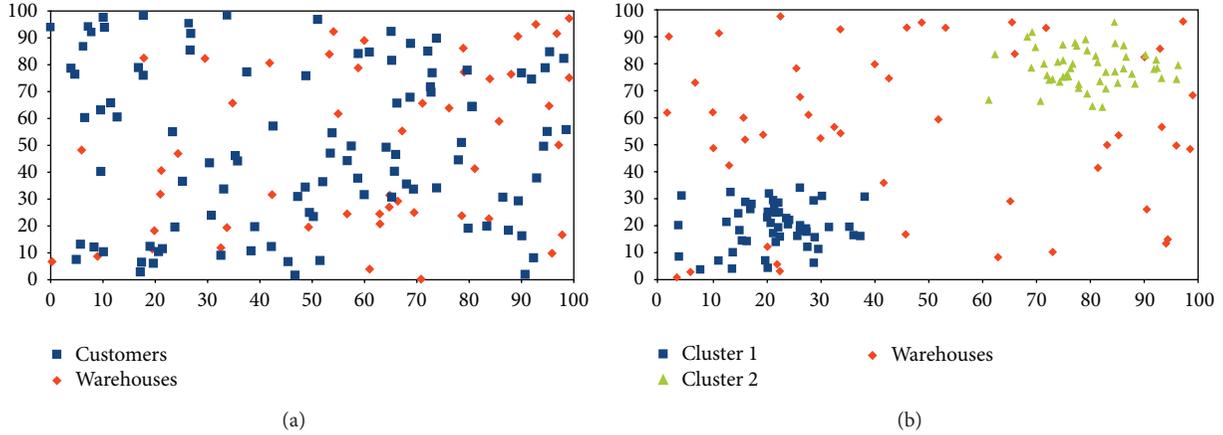


FIGURE 1: Customer and warehouse distributions. (a) Shows that customers (blue squares) are uniformly distributed. (b) Shows that customers are distributed over two clusters (blue squares and green triangles). In both cases, warehouses (orange diamonds) are uniformly distributed.

a number of iterations [41]. One distinctive feature of our algorithm is the neighbourhood definition. The idea is that we know that every particle  $p^k$  is able to see the best particle at the current iteration. Then, we assume that each particle  $k$  can see, at least, any other particle  $l$  for which the distance  $d_{k-l}$  from  $p^k$  to  $p^l$  is less than or equal to distance  $d_{k-best}$  from  $p^k$  to  $p^{best}$ . Distance  $d_{k-l}$  is calculated as follows:

$$\text{dist}_{k-l} = \sum_{j=0}^M \sqrt{(p_j^k - p_j^l)^2}. \quad (20)$$

It is clear that in this paper, we do not attempt to develop an original strategy to solve our problem. Instead, we are proposing very simple procedures to obtain solutions for our new model that can be used as a baseline in future research. Moreover, validating some of the main model assumptions is also one of the goals of this work.

## 5. Experiments and Computational Results

In this section, we present the experiments we have carried out, and we draft some conclusions about our numerical results. As we have noted before in this paper, our goal is not to state whether the TS or PSO approach is the best way to solve this problem. In fact, we firmly believe that other approaches might be more effective than our approach, and, therefore, applying different strategies to solve our model appears to be a fruitful area for future research into artificial intelligence field. In spite of this, due to the simplicity that both TS and PSO offer, we present these results as a baseline for future studies. We have selected these two algorithms as samples of local search and evolutionary strategies, respectively. Therefore, two of the most important approaches used to solve large and complex combinatorial optimisation problems are covered with these two algorithms.

Moreover, in this section, we present a set of instances that can be used as a benchmark in future studies. The benchmark set consists of two base instances, namely,  $C_1$  and  $C_2$ , which means that customers are distributed uniformly

and in clusters, respectively. Therefore, for instances of type  $C_1$  customers and DCs are randomly distributed, whereas, for instances of type  $C_2$ , only DCs are distributed over the entire area. Customers are distributed randomly along a specific perimeter around the centre of the cluster. The maximum distance allowed between a customer and the centre of the cluster is a parameter. In our case, the number of clusters was equal to 2 and the ratio was 25[km]. These two base instances ( $C_1$  and  $C_2$ ) were used for all the corresponding instances. Customer demand  $d_j$  is calculated using a uniform distribution  $U[30, 100]$ . Transportation costs (TC)  $Y_{ij}$  correspond to 1/5 of the Euclidean distance between a customer  $j$  and a warehouse  $i$ . Fixed costs (FC) are calculated as the sum of a constant value  $b$  (in our case 5000) and a factor dependent on the distance between the warehouse and its closest cluster ( $10 \times b / \text{mindist}$ ). With this expression we make those facilities that are close to the “city centre” more expensive. In the same way, the capacity of the warehouse  $I^{\text{cap}}$  will decrease as the distance to the centre of the cluster becomes small.  $I_i^{\text{cap}}$  is calculated as the sum of a base value  $c = 100$  plus  $10 \times \text{mindist}$ . The order cost (OC) and holding cost (HC) depend on the  $I^{\text{cap}}$ . The larger the capacity, the smaller the inventory costs. LT is a random number between 2 and 4. The size of the order  $Q^{\text{cap}}$  is determined as the sum of a constant equal to 100 plus a percentage of the capacity of the warehouse  $I^{\text{cap}}$ . In our case, this percentage was 50%. FC, HC, OC, Service Levels ( $Z_{1-a}$ ), LT,  $I^{\text{cap}}$ , and  $Q^{\text{cap}}$  for each DC are the same for the two base instances. Figure 1 shows these configurations.

From these two main configurations, we generated a set of 22 instances based on them (11 for each one). To do that, we modified values of FC, HC, and OC in  $\pm 25\%$ , and additionally we have varied parameter  $R$  among values 1, 2 and 3. Then, each instance attempts to evaluate the impact on the network configuration (and on its costs) produced by either increasing or decreasing the parameter values, one at a time. As base instances, we have  $C_1$  and  $C_2$  and based on them we have a set of instances such as  $C_1FC_{75}$ , which means that the FCs from base instance  $C_1$  have been reduced to 75% of their original value.

**5.1. Computational Results and Discussion.** The computational experiments were performed on an Intel Core Duo processor CPU T2700 2.33 GHz with 2 GB of RAM, using the Ubuntu 12.04 operating system. Both the TS and PSO algorithms were implemented in the JAVA programming language using NetBeans IDE. To validate the algorithms and measure their convergence, we used a set of small instances from [42] that have an optimal solution that was obtained by means of an enumerative method. This enumerative algorithm was executed for approximately 4 hours for each instance, whereas the heuristic algorithms took less than 5 seconds to solve each one. Furthermore, they found the optimal solution all 10 times that they were executed over each instance. We also ran a random search algorithm to determine the real improvement provided by our heuristic. On average, savings when using heuristics algorithms were approximately 300% in all instances.

We fixed both the nonstockout probability and the nonvolute capacity constraint probability at 95%. Hence, the values for  $\alpha$  and  $\beta$  are fixed at 1.648. As we mentioned in Section 4 of this paper, we implemented two types of neighbourhood movements for our TS algorithm. Table 1 shows a summary of the results obtained using both movements for all 22 instances.

Column *Instance* corresponds to the evaluated instance. Columns  $\overline{x_{m1}}$  and  $\overline{x_{m2}}$  correspond to the average value obtained after executing the TS algorithms 10 times using movements  $M1$  and  $M2$ , respectively. In the same way, columns  $\sigma_{m1}$  and  $\sigma_{m2}$  correspond to the standard deviation of the results. Finally, columns  $\text{Time}_{m1}$  (sec) and  $\text{Time}_{m2}$  (sec) show the average time needed by our TS implementation to find the best solution.

Table 1 shows how, predictably, the average cost increases with the value of period review  $R$ . Furthermore, the DND is also affected by the change in the value of  $R$ . In several instances, when  $R$  increases, the number of open DCs follows it. Figures 2(a) and 2(b) illustrate this situation.

These two situations are produced for the same reason: when the value of  $R$  increases, the safety stock must be greater to avoid stockout during the period without review. Thus, DCs cannot attend the same number of customers when the safety stock is increased because of their limited capacity. For this reason, the model is forced to open an extra DC to satisfy total customer demand. This relation is especially important because it shows that the value of  $R$  not only affects the cost of the distribution network but also its design.

We now compare the best solutions obtained by the TS algorithm with the ones obtained by the PSO method. Table 2 shows the results of this comparison.

According to the results the PSO algorithm seems to perform slightly better than the TS. That is more evident when we look at  $C_2$  instances where PSO shows a better performance in 9 out of 11 cases. On the other hand, TS is clearly better than PSO when  $C_1$  instances are considered. These are quite interesting results because they suggest that we should take into account how customers are distributed to decide among different heuristic techniques. It is clear that further research is needed in this field.

Because we do not have any bound for our instances, we cannot say anything about the quality of our solutions. Another interesting research line would be to find lower bounds for this model to determine how good the solutions provided by the heuristic methods are.

Moreover, when we examine the components of our objective function, we can see how they behave. In Table 3, columns FC (%), TC (%), Inv (%), and SS (%) show the change (on average) of *Location*, *Allocation*, *Inventory*, and *Safety Stock* cost as a percentage of the base instance  $C_iR_1$ .

Another interesting finding is that individual subcosts (FC, TC, Inventory, and SS) do not increase to the same extent that parameters do. In other words, when, for example, FC is increased (decreased) by 25%, the portion of the total cost corresponding to FC does not increase by 25% when compared with its corresponding base instance. This can be associated with the ability of the model to seek other location/allocation alternatives to keep the total cost as low as possible.

Regarding our algorithms, they showed good performance in terms of time and convergence. As we mentioned before, we cannot compare our results with other works because this is a very new model. Figure 3 shows the convergence of the TS algorithm for a specific instance. Here, we can see how the diversification method was invoked several times (peaks in the solid line) during the execution after finding a (presumably) local optima.

Despite the fact that restart method proved to be effective in providing some diversification to our algorithm (and consequently allowing us to move away from local optima), other diversification strategies can be implemented to exploit information from previous iterations and, consequently, provide the algorithm a higher exploration level. With regard to the *aspiration criterion*, this was invoked 30 times on average for each execution with a lower bound of 4 and an upper bound of 71. These values show how the frequency-based memory of the TS works.

## 6. Conclusions and Future Work

In this paper, we solve a novel inventory-location model that integrates inventory decisions at the strategic level. As a first contribution, our model considers a periodic inventory review policy  $(R, s, S)$  unlike previous models that assume continuous inventory review policies. We have also generated a set of 22 medium-sized instances for both *uniformly* and *2-clustered* distributed customers. These medium-sized instances consider 50 DCs and 100 customers. This set of instances was solved using both TS and PSO heuristics independently which yielded an approximation to the optimal solution for each instance. Despite the simplicity of both approaches, good solutions (w.r.t. randomly generated ones) were found in an acceptable amount of time. The results obtained confirm the importance of the value of  $R$ , which affects both the cost and configuration of the distribution network. A sensitivity analysis over the key parameters of the model was performed. The numerical results showed the effect on the network produced by changes in both inventory cost (HC and OC) and TC. As future work it is possible to

TABLE 1: Summary results. Comparison between the results obtained by movements  $m_1$  and  $m_2$ .

Instance	$\bar{x}_{m_1}$	$\sigma_{m_1}$ (%)	Time $_{m_1}$ (sec)	$\bar{x}_{m_2}$	$\sigma_{m_2}$ (%)	Time $_{m_2}$ (sec)	Gap (%)
$C_1FC_{-25}$	151541.81	1.57%	305.93	142772.05	0.69%	94.62	5.79%
$C_1FC_{+25}$	176910.48	0.76%	175.37	172204.30	0.50%	46.83	2.66%
$C_1TC_{-25}$	155167.37	7.69%	97.86	125920.71	9.85%	86.94	18.85%
$C_1TC_{+25}$	173094.71	7.54%	289.57	164272.51	9.61%	104.37	5.10%
$C_1HC_{-25}$	144888.67	8.96%	223.16	134974.87	11.98%	79.43	6.84%
$C_1HC_{+25}$	185303.83	6.05%	166.17	173147.78	7.57%	88.09	6.56%
$C_1OC_{-25}$	164188.72	1.48%	155.14	155644.07	0.67%	64.94	5.20%
$C_1OC_{+25}$	163959.14	1.39%	60.80	157637.04	0.30%	67.82	3.86%
$C_1R_1$	164081.02	7.75%	203.51	152860.21	16.25%	87.88	6.84%
$C_1R_2$	226197.19	5.71%	239.07	228169.26	11.13%	249.08	-0.87%
$C_1R_3$	300915.27	3.25%	217.26	301631.38	6.89%	198.21	-0.24%
$C_2FC_{-25}$	142217.21	0.43%	235.52	123147.56	2.31%	114.04	13.41%
$C_2FC_{+25}$	168438.53	0.33%	121.61	142471.58	1.52%	90.45	15.42%
$C_2TC_{-25}$	146509.00	4.64%	218.86	124372.47	14.23%	199.12	15.11%
$C_2TC_{+25}$	161718.90	6.10%	294.69	136471.36	15.16%	67.86	15.61%
$C_2HC_{-25}$	137153.56	7.16%	240.17	114363.85	17.62%	82.98	16.62%
$C_2HC_{+25}$	172214.43	4.57%	94.59	148850.52	11.69%	65.94	13.57%
$C_2OC_{-25}$	155132.44	1.61%	181.48	133414.40	0.83%	53.26	14.00%
$C_2OC_{+25}$	156367.93	1.39%	62.21	133270.49	0.73%	68.75	14.77%
$C_2R_1$	154938.46	9.34%	216.25	132817.52	10.40%	69.38	14.28%
$C_2R_2$	197953.21	7.59%	327.43	198309.61	7.06%	164.28	-0.18%
$C_2R_3$	267045.25	4.54%	254.97	266821.85	4.58%	259.85	0.08%

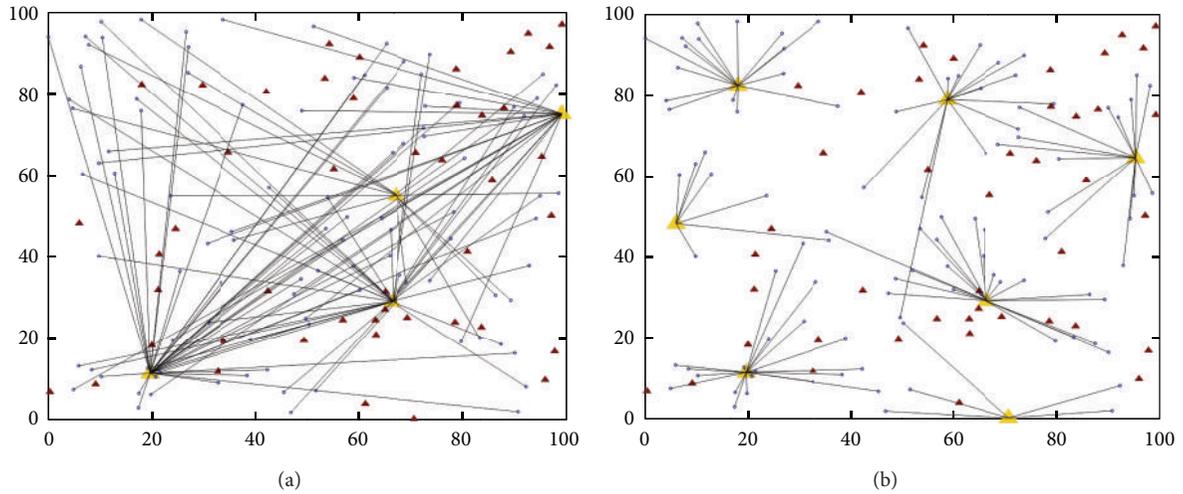


FIGURE 2: (a) Shows the DND obtained when parameter  $R = 1$  and (b) shows the DND obtained when parameter  $R = 3$ .

integrate other tactical elements such as routing or layout decisions to our ILM, to make them even more representative of reality and therefore improve the quality of the solutions. Examples of these elements are the transportation decisions or the DC layout decisions. Based on the obtained results, we can state (preliminary) that evolutionary algorithms and particularly those based on swarm intelligence such as PSO should be implemented to obtain better solutions for those instances that involve clusters. Local search strategies should

be considered for those instances with uniformly distributed customers. However, further investigation is needed in this area.

Moreover, some *matheuristics* approaches (ones that mix mathematical programming and metaheuristics) seem to be another interesting research area to explore. Finally, it would also be interesting to calculate some lower bounds for this model in order to measure how far from the optimal solution our approximations are.

TABLE 2: Comparison between results obtained by the local search (TS) and the evolutionary algorithm (PSO).

Instance	TS	PSO	Dif (%)
$C_1FC_{+25}$	142,772.05	142,849.31	0.054%
$C_1FC_{-25}$	172,204.30	166,232.44	-3.468%
$C_1TC_{-25}$	125,920.71	147,587.20	17.206%
$C_1TC_{+25}$	164,272.51	164,450.20	0.108%
$C_1HC_{-25}$	134,974.87	141,538.00	4.862%
$C_1HC_{+25}$	173,147.78	178,852.42	3.295%
$C_1OC_{-25}$	155,644.07	157,435.47	1.151%
$C_1OC_{+25}$	157,637.04	157,176.59	-0.292%
$C_1R_1$	152,860.21	156,823.91	2.593%
$C_1R_2$	226,197.19	226,450.57	0.112%
$C_1R_3$	300,915.27	301,078.22	0.054%
$C_2FC_{-25}$	123,147.56	119,074.55	-3.307%
$C_2FC_{+25}$	142,471.58	145,809.30	2.343%
$C_2TC_{-25}$	124,372.47	122,065.28	-1.855%
$C_2TC_{+25}$	136,471.36	135,172.27	-0.952%
$C_2HC_{-25}$	114,363.85	108,741.32	-4.916%
$C_2HC_{+25}$	148,850.52	142,216.14	-4.457%
$C_2OC_{-25}$	133,414.40	127,997.33	-4.060%
$C_2OC_{+25}$	133,270.49	133,320.84	0.038%
$C_2R_1$	132,817.52	119,972.02	-9.672%
$C_2R_2$	198,309.61	197,929.53	-0.192%
$C_2R_3$	266,821.85	262,059.09	-1.785%

TABLE 3: Variation of cost distribution at each instance as a percentage of base instances  $C_iR_1$ .

Instance	FC (%)	TC (%)	Inv (%)	SS (%)
$C_1FC_{-25}$	13.75%	-3.59%	-8.11%	-7.35%
$C_1FC_{+25}$	-24.85%	15.95%	10.29%	10.88%
$C_1FC_{-25}$	8.15%	16.00%	-9.89%	-10.61%
$C_1FC_{+25}$	-4.25%	-11.52%	6.07%	6.62%
$C_1FC_{-25}$	-16.76%	-11.70%	13.72%	14.86%
$C_1FC_{+25}$	9.79%	12.35%	-9.87%	-10.42%
$C_1FC_{-25}$	-5.81%	6.78%	1.70%	1.41%
$C_1FC_{+25}$	-7.10%	7.48%	1.29%	2.53%
$C_1R_1$	<b>0.00%</b>	<b>0.00%</b>	<b>0.00%</b>	<b>0.00%</b>
$C_1R_2$	31.66%	31.96%	-28.61%	-31.50%
$C_1R_3$	48.95%	47.11%	-44.14%	-47.62%
$C_2FC_{-25}$	19.18%	-7.42%	-7.93%	-7.88%
$C_2FC_{+25}$	-14.05%	2.19%	6.82%	6.87%
$C_2FC_{-25}$	2.38%	16.64%	-7.00%	-6.63%
$C_2FC_{+25}$	5.64%	-17.16%	1.94%	2.95%
$C_2FC_{-25}$	-17.46%	-8.89%	11.43%	12.80%
$C_2FC_{+25}$	13.25%	12.61%	-11.09%	-11.40%
$C_2FC_{-25}$	-2.13%	0.07%	2.11%	0.62%
$C_2FC_{+25}$	0.97%	-1.34%	-1.04%	0.41%
$C_2R_1$	<b>0.00%</b>	<b>0.00%</b>	<b>0.00%</b>	<b>0.00%</b>
$C_2R_2$	37.02%	36.05%	-30.89%	-32.30%
$C_2R_3$	52.70%	53.47%	-44.24%	-46.91%

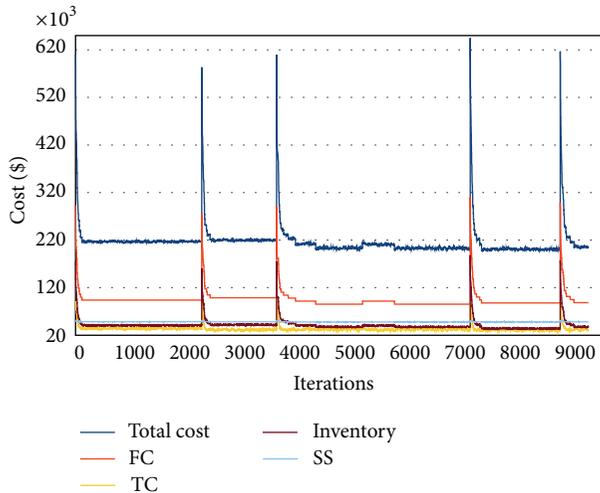


FIGURE 3: Current solution costs. The peaks correspond to the values of the current solution when the restart method is invoked.

## Acknowledgment

The authors are indebted to anonymous referee comments and the editor for their valuable comments through the review process. The paper remarkably improved through their recommendations; yet, the authors are responsible for any remaining errors. Fernando Paredes is supported by FONDECYT-Chile Grant 1130455.

## References

- [1] D. Simchi-Levi and Y. Zhao, "The value of information sharing in a two-stage supply chain with production capacity constraints," *Naval Research Logistics*, vol. 50, no. 8, pp. 888–916, 2003.
- [2] M. Mourits and J. J. Evers, "Distribution network design: an integrated planning support framework," *International Journal of Physical Distribution Logistics Management*, vol. 25, pp. 43–57, 1995.
- [3] J. R. Bradley and B. C. Arntzen, "The simultaneous planning of production, capacity, and inventory in seasonal demand environments," *Operations Research*, vol. 47, no. 6, pp. 795–806, 1999.
- [4] P. A. Miranda and R. A. Garrido, "Incorporating inventory control decisions into a strategic distribution network design model with stochastic demand," *Transportation Research E*, vol. 40, no. 3, pp. 183–207, 2004.
- [5] P. Miranda, *Un enfoque integrado para el diseño estratégico de redes de distribución de carga [Ph.D. thesis]*, Pontificia Universidad Católica de Chile, 2004.
- [6] M. S. Daskin, *Network and Discrete Location: Models, Algorithms, and Applications*, Wiley-Interscience, New York, NY, USA, 1st edition, 1995.
- [7] D. Simchi-Levi, X. Chen, and J. Bramel, *The Logic of Logistics*, Springer, New York, NY, USA, 2005.
- [8] Z. Drezner and H. Hamacher, *Facility Location. Applications and Theory*, Springer, Berlin, Germany, 2002.
- [9] M. S. Daskin, C. R. Coullard, and Z.-J. M. Shen, "An inventory-location model: formulation, solution algorithm and computational results," *Annals of Operations Research*, vol. 110, pp. 83–106, 2002.
- [10] Z.-J. M. Shen, C. Coullard, and M. S. Daskin, "A joint location-inventory model," *Transportation Science*, vol. 37, no. 1, pp. 40–55, 2003.
- [11] P. A. Miranda and R. A. Garrido, "Valid inequalities for Lagrangian relaxation in an inventory location problem with stochastic capacity," *Transportation Research E*, vol. 44, no. 1, pp. 47–65, 2008.
- [12] L. Ozsen, C. R. Coullard, and M. S. Daskin, "Capacitated warehouse location model with risk pooling," *Naval Research Logistics*, vol. 55, no. 4, pp. 295–312, 2008.
- [13] S. K. Kumar and M. Tiwari, "Supply chain system design integrated with risk pooling," *Computers Industrial Engineering*, vol. 64, pp. 580–588, 2013.
- [14] J.-S. Tancrez, J.-C. Lange, and P. Semal, "A location-inventory model for large three-level supply chains," *Transportation Research E*, vol. 48, no. 2, pp. 485–502, 2012.
- [15] Z. Firoozi, N. Ismail, Sh. Ariaifar, S. H. Tang, M. K. A. M. Ariffin, and A. Memariani, "Distribution network design for fixed lifetime perishable products: a model and solution approach," *Journal of Applied Mathematics*, vol. 2013, Article ID 891409, 13 pages, 2013.
- [16] O. Berman, D. Krass, and M. M. Tajbakhsh, "A coordinated location-inventory model," *European Journal of Operational Research*, vol. 217, no. 3, pp. 500–508, 2012.
- [17] J. F. Bard and N. Nananukul, "A branch-and-price algorithm for an integrated production and inventory routing problem," *Computers & Operations Research*, vol. 37, no. 12, pp. 2202–2217, 2010.
- [18] H. Badri, M. Bashiri, and T. H. Hejazi, "Integrated strategic and tactical planning in a supply chain network design with a heuristic solution method," *Computers & Operations Research*, vol. 40, no. 4, pp. 1143–1154, 2013.
- [19] V. A. Armentano, A. L. Shiguemoto, and A. Løkketangen, "Tabu search with path relinking for an integrated production-distribution problem," *Computers & Operations Research*, vol. 38, no. 8, pp. 1199–1209, 2011.
- [20] R. G. Askin, I. Baffo, and M. Xia, "Multi-commodity warehouse location and distribution planning with inventory consideration," *International Journal of Production Research*, 2013.
- [21] P. A. Miranda and R. A. Garrido, "A simultaneous inventory control and facility location model with stochastic capacity constraints," *Networks and Spatial Economics*, vol. 6, no. 1, pp. 39–53, 2006.
- [22] A. Charnes and W. W. Cooper, "Chance-constrained programming," *Management Science*, vol. 6, pp. 73–79, 1959/1960.
- [23] S. Axsater, "Approximate optimization of a two-level distribution inventory system," *International Journal of Production Economics*, vol. 81–82, pp. 545–555, 2003, Proceedings of the 7th International Symposium on Inventories.
- [24] F. Glover and M. Laguna, *Tabu Search*, Kluwer Academic, Norwell, Mass, USA, 1997.
- [25] M. Pirlot, "General local search methods," *European Journal of Operational Research*, vol. 92, no. 3, pp. 493–511, 1996.
- [26] V. Vails, M. A. Perez, and M. S. Quintanilla, "A tabu search approach to machine scheduling," *European Journal of Operational Research*, vol. 106, no. 2–3, pp. 277–300, 1998.
- [27] S. Hanafi and A. Freville, "An efficient tabu search approach for the 0-1 multidimensional knapsack problem," *European Journal of Operational Research*, vol. 106, no. 2–3, pp. 659–675, 1998.

- [28] J. Brandão and A. Mercer, "A tabu search algorithm for the multi-trip vehicle routing and scheduling problem," *European Journal of Operational Research*, vol. 100, no. 1, pp. 180–191, 1997.
- [29] K. S. Al-Sultan and M. A. Al-Fawzan, "A tabu search approach to the uncapacitated facility location problem," *Annals of Operations Research*, vol. 86, pp. 91–103, 1999.
- [30] G. Guillermo Cabrera, E. Cabrera, R. Soto, L. J. M. Rubio, B. Crawford, and F. Paredes, "A hybrid approach using an artificial bee algorithm with mixed integer programming applied to a large-scale capacitated facility location problem," *Mathematical Problems in Engineering*, vol. 2012, Article ID 954249, 14 pages, 2012.
- [31] L. Dupont, "Branch and bound algorithm for a facility location problem with concave site dependent costs," *International Journal of Production Economics*, vol. 112, no. 1, pp. 245–254, 2008, Special Section on Recent Developments in the Design, Control, Planning and Scheduling of Productive Systems.
- [32] M. A. Arostegui Jr., S. N. Kadipasaoglu, and B. M. Khumawala, "An empirical comparison of Tabu Search, Simulated Annealing, and Genetic Algorithms for facilities location problems," *International Journal of Production Economics*, vol. 103, no. 2, pp. 742–754, 2006.
- [33] R. Eberhart, P. Simpson, and R. Dobbins, *Computational Intelligence PC Tools*, Academic Press, San Diego, Calif, USA, 1996.
- [34] R. Eberhart and J. Kennedy, "New optimizer using particle swarm theory," in *Proceedings of the 6th IEEE International Symposium on Micro Machine and Human Science*, pp. 39–43, October 1995.
- [35] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, vol. 4, pp. 1942–1948, December 1995.
- [36] J. Kennedy and R. C. Eberhart, *Swarm Intelligence*, Morgan Kaufmann, San Francisco, Calif, USA, 2001.
- [37] J. Kennedy, "Particle swarm: social adaptation of knowledge," in *Proceedings of the IEEE International Conference on Evolutionary Computation (ICEC '97)*, pp. 303–308, April 1997.
- [38] G. Guillermo Cabrera, D. Silvana Roncagliolo, J. P. Riquelme, C. Cubillos, and R. Soto, "A hybrid particle swarm optimization-simulated annealing algorithm for the probabilistic travelling salesman problem," *Studies in Informatics and Control*, vol. 21, pp. 49–58, 2012.
- [39] L. Cagnina, S. Esquivel, and C. A. Coello, "Hybrid particle swarm optimizers in the single machine scheduling problem: an experimental study," *Studies in Computational Intelligence*, vol. 49, pp. 143–164, 2007.
- [40] R. Poli, *Analysis of the Publications on the Applications of Particle Swarm Optimisation*, *Journal of Artificial Evolution and Applications*, vol. 2008, Article ID 685175, 14 pages, 2008.
- [41] *Empirical Study of Particle Swarm Optimization*, vol. 3, 1999.
- [42] P. A. Miranda and G. Guillermo Cabrera, "Inventory location problem with stochastic capacity constraint under periodic review (r, s, s)," in *International Conference on Industrial Logistic (ICIL '10)*, vol. 1, pp. 289–296, 2010.

## Research Article

# Genetic Pattern Search and Its Application to Brain Image Classification

Yudong Zhang,<sup>1</sup> Shuihua Wang,<sup>1,2</sup> Genlin Ji,<sup>1</sup> and Zhengchao Dong<sup>3</sup>

<sup>1</sup> School of Computer Science and Technology, Nanjing Normal University, Nanjing, Jiangsu 210023, China

<sup>2</sup> School of Electronic Science and Engineering, Nanjing University, Nanjing, Jiangsu 210046, China

<sup>3</sup> Brain Imaging Laboratory & MRI Unit, Columbia University and New York State Psychiatric Institute, New York, NY 10032, USA

Correspondence should be addressed to Yudong Zhang; zhangyudongnuaa@gmail.com

Received 1 July 2013; Revised 20 August 2013; Accepted 7 September 2013

Academic Editor: Vishal Bhatnagar

Copyright © 2013 Yudong Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A novel global optimization method, based on the combination of genetic algorithm (GA) and generalized pattern search (PS) algorithm, is proposed to find global minimal points more effectively and rapidly. The idea lies in the facts that GA tends to be quite good at finding generally good global solutions, but quite inefficient in finding the last few mutations for the absolute optimum, and that PS is quite efficient in finding absolute optimum in a limited region. The novel algorithm, named as genetic pattern search (GPS), employs the GA as the search method at every step of PS. Experiments on five different classical benchmark functions (consisting of Hump, Powell, Rosenbrock, Schaffer, and Woods) demonstrate that the proposed GPS is superior to improved GA and improved PS with respect to success rate. We applied the GPS to the classification of normal and abnormal structural brain MRI images. The results indicate that GPS exceeds BP, MBP, IGA, and IPS in terms of classification accuracy. This suggests that GPS is an effective and viable global optimization method and can be applied to brain MRI classification.

## 1. Introduction

The evolutionary computation community has shown for many years significant interest in optimization problems, in particular in the global optimization of numerical, real-valued “black-box” problems, for which exact and analytical methods are not productive. Genetic algorithm (GA) [1], generalized pattern search (PS) [2], particle swarm optimization (PSO) [3], firefly algorithm [4], and differential evolution (DE) [5] are among the most recent developments. These techniques have shown great promise in several real-world applications.

In most cases, an optimization problem is divided into two phases: the coarse search and the fine search [6]. GA is well suited for a swift and global exploration of a large search space to optimize an objective function and to target the region near the optimum point quickly [7]. However, GA may run with difficulty in the immediate vicinity of the optimum point [8]. Conversely, PS is a nonrandom method to search for minima of a function that is not differentiable,

or even continuous without requiring the gradient information [9]. It performs especially well in local search, but it is sensitive for the randomly or manually input initial values, and requires a high degree of expertise by the user [10].

The complementary advantages of GA and PS motivated our strategy in this paper that combines both GA and PS to produce a new algorithm referred to as genetic pattern search (GPS). We evaluated the proposed method by five different classical benchmark functions (consisting of Hump, Powell, Rosenbrock, Schaffer, and Woods) and applied it to the classification of normal and abnormal brain MRI images.

The structure of this paper was organized as follows. Section 2 gave a detailed introduction to GA and PS, respectively. Section 3 outlined the structure and flow of the proposed GPS algorithm. Experiments in Section 4 compared GPS with improved GA and improved PS on five test functions. Section 5 applied the GPS to structural brain image classification. Finally Section 6 concluded this paper.

## 2. Background

**2.1. Principles of GA.** GAs are powerful stochastic search techniques based on the processes of natural selection [11]. These techniques perform heuristic search that mimics the process of natural evolution, such as inheritance, mutation, selection, and crossover. This heuristic is routinely used to generate useful solutions to optimization and search problems [12]. The principles of the GAs can be described as follows.

The trial solutions of GAs are encoded in the form of strings. Each string is associated with an objective function that represents the degree of the fitness of the string. A collection of such strings is called a population. A random population with a few strings with higher fitness is initially created to represent different points in the search space. Each of these strings is assigned a number of copies that go into the mating pool, based on the principle of survival of the fittest [13]. Crossover and mutation operators are applied on these strings. The processes of selection, crossover, and mutation continue until either a fixed number of generations or a termination condition is reached [14]. The above procedures of GA can be realized by the following pseudocode.

- (1) Choose the initial population of individuals.
- (2) Evaluate the fitness of each individual in that population.
- (3) Repeat on this generation until termination criteria are met.
  - (A) Select the best-fit individuals for reproduction.
  - (B) Breed new individuals through crossover and mutation operations to give birth to offspring.
  - (C) Evaluate the individual fitness of new individuals.
  - (D) Replace the least-fit population with new individuals.

GAs differ from classical optimization techniques such as the gradient-based algorithm in the following three ways: (1) GAs work on a population of points instead of a single point; (2) GAs use only the values of the objective function not their derivatives or other auxiliary knowledge; (3) GAs use probabilistic transition functions and not deterministic ones [15].

**2.2. Introduction of PS.** PS is a method that updates current iterate by sampling the objective function to find a decrease at a finite number of points along a suitable set of search directions [16]. Suppose that;  $f$  denotes the objective function, starting from an initial guess  $x_0$  and initial step length  $\Delta_0$ , the PS generates a sequence of iterates such that  $f(x_{k+1}) \leq f(x_k)$ . Each iteration consists of a “search” step and a “poll” step. The search step generates a finite number of trial points on a mesh  $M_k$ , which is centered at  $x_k$  and defined by a finite set of patterns  $\Gamma$ , which positively span the solution space. The mesh is given by

$$M_k(i) = \{x_k + \Delta_k d_i, d_i \in \Gamma\}. \quad (1)$$

The poll step polls the points in the current mesh to find a better one. The polling can be either complete, meaning that all points are polled, or incomplete, namely, the algorithm stops polling as soon as it finds a point whose objective function value is less than the current point. The complete poll performs better but consumes more time. The incomplete poll finds the local optima [17].

If the poll step finds an improved point, then  $x_{k+1}$  equals to the new point, and update the step length by multiplying  $\Delta_k$  by expansion factor  $\alpha_e (>1)$  such that  $\Delta_{k+1} > \Delta_k$  because current pattern is a suitable set of poll directions. Otherwise, the poll step cannot find an improved point; then  $x_{k+1} = x_k$ , and the step length is reduced by multiplying  $\Delta_k$  by contraction factor  $\alpha_c (<1)$  such that  $\Delta_{k+1} < \Delta_k$ . The aforementioned description can be summarized as follows.

- (1) Initialization: generate patterns  $\Gamma$  and initialize the step length  $\Delta_0$ .
- (2) Repeat on this generation until termination criteria are met:
  - (A) generate new mesh points  $\{M_k\}$ ;
  - (B) poll the mesh points. If successful, expand the mesh; otherwise contract the mesh.

## 3. Genetic Pattern Search

The proposed genetic pattern search (GPS) combines both GA and PS algorithms. In 2011, Zhang et al. have proposed a combination based on GA and PS [18]; however, in their method the combination algorithm was done by a first stage using a GA followed by a second stage that uses PS taking as an input the output of the GA. The method has a serious drawback: the PS could start when the GA has not reached the neighborhood of the global optimum yet. In that way, we would not take full advantage of the GA. This drawback will become more critical when the problem to be solved becomes more complex.

Therefore, in this paper, we take the GA as the search method at every step of PS. In this way, the coarse and fine search would be made in each epoch, and it could speed the convergence up as shown in Figure 1. The pseudocodes of GPS are described as follows.

Step 1 (initialization). Generate pattern  $\Gamma = \{d_1, d_2, d_3, \dots, d_{2N}\}$ , initial point  $x_0$ , initialize step length  $\Delta_0$ , and let  $k = 0$ .

Step 2. Generate new mesh points  $\{M_k(i)\} = x_k + \Delta_k \Gamma$ .

Step 3. GA Search with initial population designed as  $\{M_k(i)\}$ . The mesh points are updated as the final GA results  $\{M_k^U(i)\}$ .

Step 4 (complete poll). If  $i^*$  exists so that  $f[M_k^U(i^*)] < f(x_k)$ , then it is a successful poll. Let  $x_{k+1} = M_k^U(i^*)$ , and  $\Delta_{k+1} = \alpha_e \Delta_k$ . Otherwise, the poll cannot find an improved point; let  $x_{k+1} = x_k$ , and  $\Delta_{k+1} = \alpha_c \Delta_k$ .

Step 5 ( $k = k + 1$ ). Check whether the termination conditions are satisfied. If yes, output  $x_k$ , otherwise jump to Step 2.

TABLE 1: Information of five benchmark functions.

Test function	Function formulation	Optimal point
Hump	$4x_1^2 - 2.1x_1^4 + x_1^6/3 + x_1x_2 - 4x_2^2 + 4x_2^4 + 2$	$F(-0.0898, 0.7126) = 0.9684$ $F(0.0898, -0.7126) = 0.9684$
Powell	$(x_1 + 10x_2)^2 + 5(x_3 - x_4)^2 + (x_2 - x_3)^4 + 10(x_1 - x_4)^4$	$F(0, 0, 0, 0) = 0$
Rosenbrock	$100(x_1^2 - x_2)^2 + (1 - x_1)^2$	$F(1, 1) = 0$
Schaffer	$\frac{[\sin(x_1 + x_2)^{1/2}]^2 - 0.5}{[1 + 0.001(x_1^2 + x_2^2)]^2} + 1.5$	$F(0, 0) = 1$
Woods	$100(x_2 - x_1^2)^2 + (1 - x_1)^2 + 90(x_4 - x_3)^2 + (1 - x_3)^2 + 10.1[(x_2 - 1)^2 + (x_4 - 1)^2] + 19.8(x_2 - 1)(x_4 - 1)$	$F(1, 1, 1, 1) = 0$

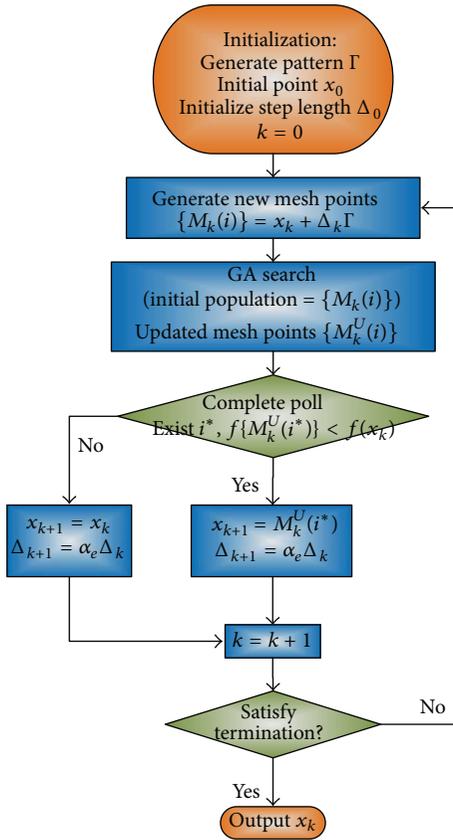


FIGURE 1: Flowchart of GPS algorithm.

## 4. Experiments

The test and evaluation were performed with an IBM P4 with 2 GHz processor and 1 GB memory. The program is in-house developed by MATLAB 2010b. The readers can reproduce the work at any laptop installing MATLAB.

**4.1. Test Functions.** We used five classical benchmark functions to evaluate the performance of the GPS algorithm. Among the five benchmark functions, Hump, Rosenbrock, and Schaffer functions are two-dimensional and Powell and Woods functions are four-dimensional. Formulations, global

optimal points, and fitness values of those functions are listed in Table 1.

It should be noted that the constant terms are added (2 in Hump function and 1.5 in Schaffer function) to make sure that the range of fitness value is always above zero so that the  $y$ -axis can be plotted in logarithmic scale. Furthermore, the initial points of each test function were selected far from the optimal point deliberately to test the robustness of the algorithms. Figure 2 shows the surface plot and contour lines of Hump, Rosenbrock, and Schaffer functions, respectively. The Powell and Woods functions are not displayed due to their high-dimensional property.

**4.2. Parameter Setting.** An improved GA (IGA) [19] and an improved PS (IPS) [20] were chosen for comparison. The parameters of IGA and IPS were obtained by numerous experiments, and the parameters corresponding to the best results are selected and listed in the first and second rows in Table 2. Besides, IGA and IPS are both evolutionary computations, which features in robustness of results with variation of parameters. Coelho et al. [21] and Ghanbari and Mahdavi-Amiri [22] have already proved that the results of evolutionary algorithms are sensitive to neither initial values nor parameter values. Therefore, the results of both IGA and IPS will exhibit little variation even if their corresponding parameters are changed far from the values determined in this experiment. The parameters of GPS were set by combining the parameters of IGA and IPS.

The initial population of IGA was set randomly distributed in the whole search space. For the deterministic algorithm IPS, we determine the initial point changed randomly in each run. In this experiment, we set the termination criteria as in the last row of Table 2 by trial-and-error method.

**4.3. Success Rate Comparison.** The success rate is the rate at which the algorithm converges to the nearly absolute global optimal point. Suppose that  $x^*$  denotes the global optimal point, and  $x$  denotes the point found in a run. A more accurate definition of success rate is shown as follows:

$$\text{if } \frac{|x - x^*|}{\max(1, |x^*|)} < 0.001, \quad \text{then } x \text{ is success.} \quad (2)$$

The above formula means that the found solution  $x$  is treated identical to the global optima  $x^*$  if the norm of their

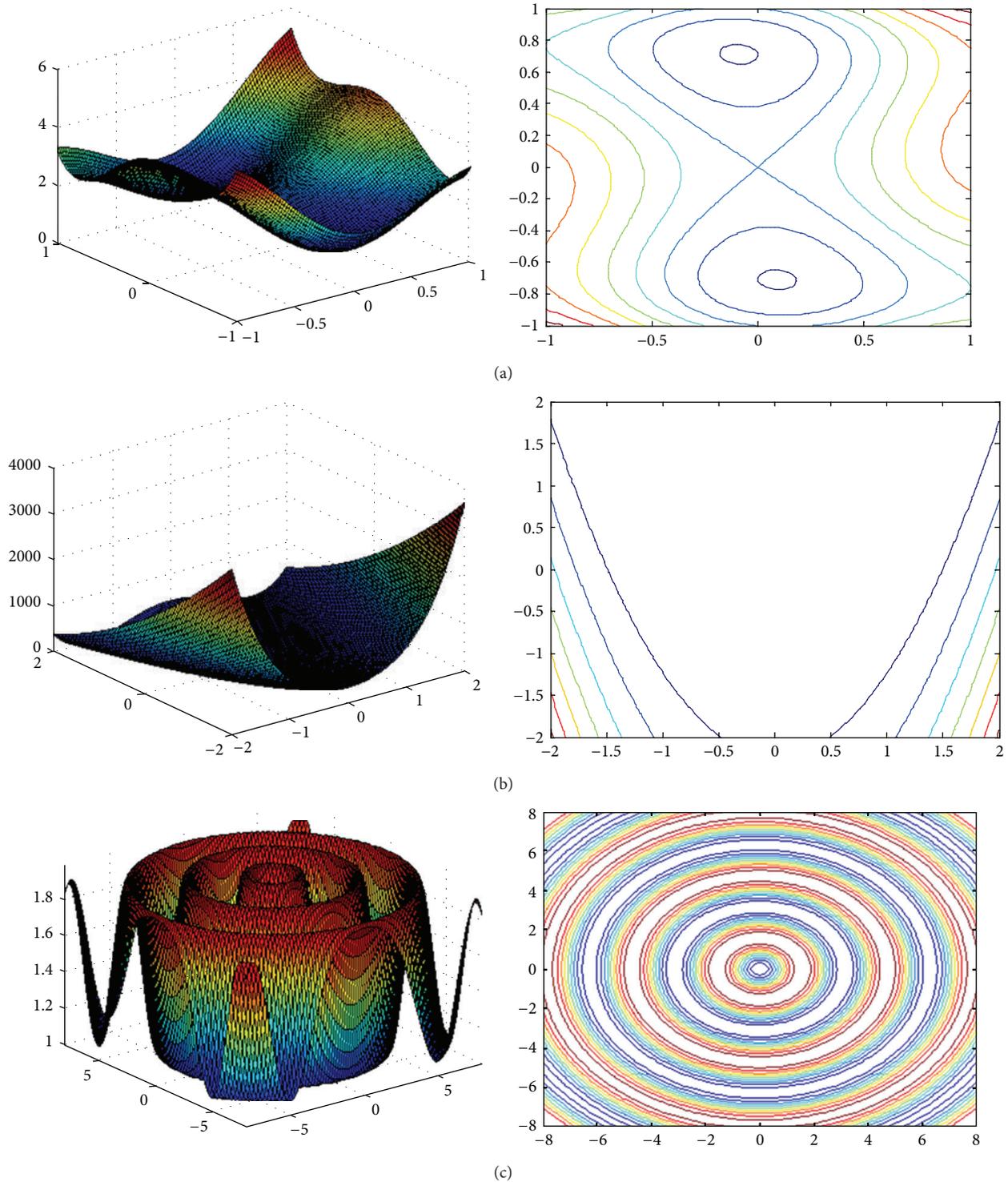


FIGURE 2: Surface plot and contour lines of 2D test functions: (a) Hump; (b) Rosenbrock; (c) Schaffer.

differences is less than 0.1% of the maximum between 1 and the norm of  $x^*$ . The reason why we do not use the rigid equation  $x = x^*$  is that finding the accurate global optimal is impossible. The reason lies in the following 3 points: (I) there are always round-up errors during the computation; (II) the word length of the computer is limited; (III) 0.1% relative error is enough in most of industrial applications.

After 100 runs of each algorithm, the success rate was calculated and listed in Table 3. The data here is not the same from [18] because the parameters are different: (I) here we set the maximum fitness evaluation as infinity while in [18] it is only  $2 \times 10^4$ ; (II) here the GPS is done by adding GA at each step of PS, while the algorithm is done by first GA followed by PS in [18]; (III) here the initialization of IPS is randomly

TABLE 2: Parameters setting.

IGA	Population = 20; elite number = 2; crossover rate = 0.8, mutation rate = 0.2, migration interval = 20; migration direction = forward; fitness scaling = rank; selection function = stochastic
IPS	Population = 2N; poll = 2N positive basis; complete poll = yes; initial mesh size = 1; expansion factor = 2.0; contraction factor = 0.5.
GPS	GA: population = 20; elite number = 2; crossover rate = 0.8, mutation rate = 0.2, fitness scaling = rank; selection function = stochastic. PS: population = 2N; Poll = 2N positive basis; complete poll = Yes; initial mesh size = 1; expansion factor = 2.0; contraction factor = 0.5.
Termination	Maximum fitness evaluate = $\infty$ ; independent variables tolerance = $10^{-4}$ ; fitness tolerance = $10^{-4}$ ; stall iteration = 20; mesh precision = $10^{-4}$ .

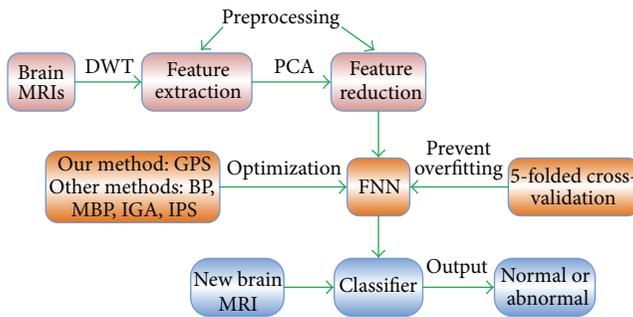


FIGURE 3: Flowchart of our brain classification system.

TABLE 3: Success rate comparison.

Test function	IGA	IPS	GPS
Hump	82	95	<b>98</b>
Powell	41	94	<b>99</b>
Rosenbrock	62	88	<b>95</b>
Schaffer	75	71	<b>96</b>
Woods	33	29	<b>85</b>

created at each run, while the initial values of IPS are fixed in [18].

For all functions, the GPS performs best with high enough success rates. For Hump, Powell, and Rosenbrock functions, the IPS is better than IGA. For Schaffer and Woods functions, the IGA is better than IPS. In total, the GPS is the most robust algorithm among the three.

## 5. Application

As an application, we applied the GPS algorithm to the weights optimization of forward neural network (FNN), which is used as a classifier of structural MRI images between normal and abnormal brains.

**5.1. Method.** The strategy is shown in Figure 3. First, we extract features via discrete wavelet transform (DWT), which has already proved to be an effective strategy for clinical diagnosis [23–25]. Second, the wavelet domain features are reduced via principle component analysis (PCA). Third, we use stratified  $K$ -fold cross-validation to prevent overfitting

of the following classifier. Fourth, the reduced features are sent to the FNN. Fifth, the GPS and other training algorithms are employed to train the FNN. Finally, the classification accuracies of FNNs trained by different algorithms are compared. The following paragraphs will discuss the procedures in detail.

The DWT is a powerful implementation of the wavelet transform using the dyadic scales and positions [26]. In this study, since the brain images are 2D data, the DWT is applied to horizontal and vertical dimensions separately. As a result, there are 4 subband images at each scale. The subband  $cA_{j+1}$  is used for next 2D DWT. As the level of decomposition increased, a more compact but coarser approximation component was obtained. Thus, wavelets provide a simple hierarchical framework for interpreting the image information.

PCA is an efficient tool to reduce the dimension of a data set consisting of a large number of interrelated variables while retaining most of the variations [27]. The PCA describes the space of the original data projecting onto the space in a base of eigenvectors. The corresponding eigenvalues account for the energy of the process in the eigenvector directions. It is assumed that most of the information in the observation vectors is contained in the subspace spanned by the first several PCs.

Cross-validation methods consist of three types: random subsampling,  $K$ -fold cross-validation, and leave-one-out validation. The  $K$ -fold cross-validation is applied due to its properties as being simple to learn, easy to realize, and using all data for training and validation. The mechanism is to create a  $K$ -fold partition of the whole dataset, repeat  $K$  times to use  $K - 1$  folds for training and a left fold for validation, and finally average the error rates of  $K$  experiments. The  $K$  folds can be randomly partitioned. However, some folds may have quite different distributions from other folds. Therefore, stratified  $K$ -fold cross-validation was employed, with which every fold has nearly the same class distributions [28]. In this study, we empirically determined  $K$  as 5 through the trial-and-error method.

FNNs are widely used in pattern classification since they do not need any information about the probability distribution and the a priori probabilities of different classes [29]. A single-hidden-layer backpropagation (BP) neural network is adopted with sigmoid neurons in the hidden layer and linear neuron in the output layer.

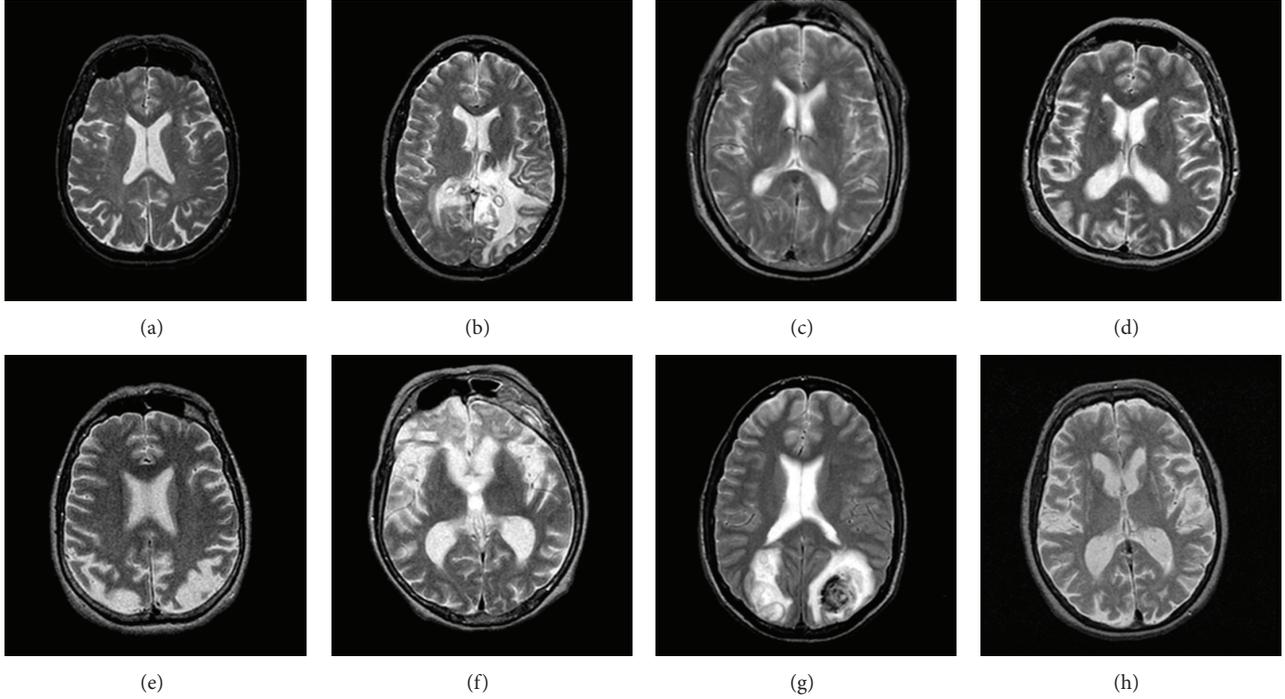


FIGURE 4: Sample of normal and abnormal brain images: (a) normal brain; (b) glioma; (c) meningioma; (d) Alzheimer's disease; (e) Alzheimer's disease with visual agnosia; (f) Pick's disease; (g) sarcoma; (h) Huntington's disease.

The training vectors were presented to the FNN, which is trained in batch mode. The network configuration is supposed as  $N_I \times N_H \times N_O$ , that is, a two-layer network with  $N_I$  input neurons,  $N_H$  neurons in the hidden layer, and  $N_O$  output indicating the brain is normal or abnormal. Assume that  $\omega_1$  and  $\omega_2$  represent the connection weight matrix between the input layer and hidden layer, between the hidden layer and the output layer, respectively, the outputs of all neurons in the hidden layer are calculated by

$$y_j = f_H \left( \sum_{i=1}^{N_I} \omega_1(i, j) x_i \right), \quad j = 1, 2, \dots, N_H. \quad (3)$$

Here,  $x_i$  denotes the  $i$ th input value,  $y_j$  denotes the  $j$ th output of the hidden layer, and  $f_H$  refers to the activation function of hidden layer, usually a sigmoid function. The outputs of all neurons in the output layer are given as follows:

$$O_k = f_O \left( \sum_{j=1}^{N_H} \omega_2(j, k) y_j \right), \quad k = 1, 2, \dots, N_O. \quad (4)$$

Here,  $f_O$  denotes the activation function of output layer, usually a line function. All weights are assigned to random values initially and are modified by the delta rule according to the learning samples. The error is expressed as the mean squared error (MSE) of the difference between output and target values:

$$E_l = \text{mse} \left( \sum_{k=1}^{N_O} (O_k - T_k) \right), \quad l = 1, 2, \dots, N_S, \quad (5)$$

where  $T_k$  represents the  $k$ th value of the authentic values, which are already known to users, and  $N_S$  represents the number of samples. Suppose that there are  $N_S$  samples, the fitness value is written as

$$F(\omega) = \sum_{l=1}^{N_S} E_l, \quad (6)$$

where  $\omega$  represents the vectorization of the  $(\omega_1, \omega_2)$ . Our goal is to minimize this fitness function  $F(\omega)$ , namely, to force the output values of each sample approximate to corresponding target values. At this point, we use our proposed method GPS to optimize formula (6), compared with other algorithms including BP [30], momentum BP (MBP) [31], IGA, and IPS.

**5.2. Simulations and Results.** The datasets consist of T2-weighted MR brain images in axial plane and  $256 \times 256$  in-plane resolution, which were downloaded from the Harvard Medical School website (<http://www.med.harvard.edu/AANLIB/>). We randomly selected 80 images consisting of 40 normal and 40 abnormal. The abnormal brain MR images consist of the following diseases: glioma, meningioma, Alzheimer's disease, Alzheimer's disease plus visual agnosia, Pick's disease, sarcoma, and Huntington's disease. A sample of each is shown in Figure 4.

Figure 5 give the three-level decomposition of 2D DWT decomposition result on a normal brain image. The upper-left corner in Figure 5(b) shows the approximate coefficients serving as the reduced features. The dimension of original image is  $256 \times 256 = 65536$ , while the dimension of approximation image is only  $32 \times 32 = 1024$ .

TABLE 4: Averaged classification accuracy (20 runs).

Algorithm	Classification accuracy
BP	58.813%
MBP	61.687%
IGA	90.063%
IPS	90.625%
GPS	<b>95.188%</b>

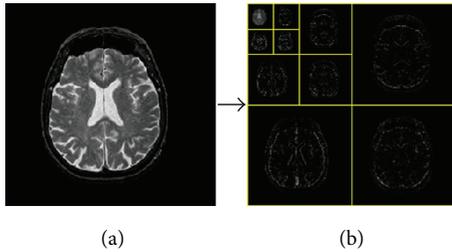


FIGURE 5: A sample of 3-level 2D DWT: (a) a normal MRI brain image; (b) 3-level wavelet coefficients.

Although the dimensions of extracted features are reduced from 65536 to 1024, this is still a high computation cost. Thus, PCA is used to further reduce the dimensions of features on another level. The curve of cumulative sum of variance with number of principle components is shown in Figure 6. It shows that 19 principle components, which are only 1.855% out of the 1024 features, preserve 95.4% of the total variance.

Those 19 principal components are submitted to the FNN. The proposed GPS method is used to optimize the weights/biases of FNN. Meanwhile, other five methods are also utilized to make a comparison. In order to reduce the randomness, each algorithm was performed 20 times. The averaged classification accuracy of each algorithm is shown in Table 4.

Table 4 indicates that the strategy based on the proposed GPS algorithm obtained higher classification accuracy (95.188%). The IPS and IGA perform nearly the same with classification accuracies about 90%. The MBP and BP algorithms performed the worst with accuracies around 60%. Here, we do not divide the dataset into training and test subsets because we have already employed the  $k$ -fold cross-validation method to avoid overfitting.

## 6. Conclusions and Discussions

Our contributions can be summarized in the following three aspects. First, we proposed a novel algorithm—GPS, for efficient and rapid global search of minimum points. It improved the robustness of pattern search and improved the convergence speed of genetic algorithm. Second, the test and evaluation on five benchmark functions further demonstrate that the proposed GPS is the most robust among the three algorithms. Third, as an example of application, we employed the GPS to the classification of normal and abnormal brain

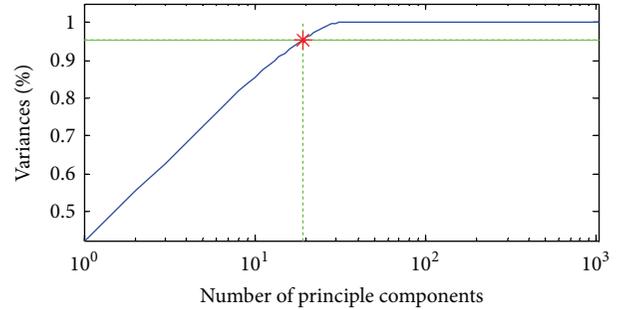


FIGURE 6: Variances against number of principle components (x-axis is log scale).

MRI images. The results indicate that GPS is superior to BP, MBP, IGA, and IPS in terms of classification accuracy.

The proposed GPS and the one in [18] are based on the combination of GA and PS. They both make use of the coarse-searching ability of GA and fine-searching ability of PS. Nevertheless, the combining principles are distinct. A question is raised as “which one is better?” Here, we give an assumption that GPS in [18] has a drawback that the PS could start when the GA has not reached the neighborhood of the global optimum yet. GPS in this paper takes the GA as the search method at every step of PS, so it will be more stable but cost more time.

In the future, we will develop simulation experiments to address the question, comparing them in convergence rate, success rate, and computation cost. We will also cooperate with mathematicians to find a theoretical solution. The future work also includes applying GPS to other academic and industrial fields.

## Acknowledgments

The authors express the sincere gratitude to the editor Dr. Vishal Bhatnagar for his careful work and the three anonymous reviewers for their valuable comments.

## References

- [1] G. Corriveau, R. Guilbault, and A. Tahan, “Genetic algorithms and finite element coupling for mechanical optimization,” *Advances in Engineering Software*, vol. 41, no. 3, pp. 422–426, 2010.
- [2] M. Wetter and E. Polak, “Building design optimization using a convergent pattern search algorithm with adaptive precision simulations,” *Energy and Buildings*, vol. 37, no. 6, pp. 603–612, 2005.
- [3] Y. Zhang, L. Wu, and S. Wang, “UCAV path planning by fitness-scaling adaptive chaotic particle swarm optimization,” *Mathematical Problems in Engineering*, vol. 2013, Article ID 705238, 9 pages, 2013.
- [4] Y. Zhang, L. Wu, and S. Wang, “Solving two-dimensional HP model for firefly algorithm and simplified energy function,” *Mathematical Problems in Engineering*, vol. 2013, Article ID 398141, 9 pages, 2013.

- [5] L. Moreno, S. Garrido, D. Blanco, and M. L. Muñoz, "Differential evolution solution to the SLAM problem," *Robotics and Autonomous Systems*, vol. 57, no. 4, pp. 441–450, 2009.
- [6] J. Verboomen, D. Van Hertem, P. H. Schavemaker et al., "Phase shifter coordination for optimal transmission capacity using particle swarm optimization," *Electric Power Systems Research*, vol. 78, no. 9, pp. 1648–1653, 2008.
- [7] Y. Kuroki, G. S. Young, and S. E. Haupt, "UAV navigation by an expert system for contaminant mapping with a genetic algorithm," *Expert Systems with Applications*, vol. 37, no. 6, pp. 4687–4697, 2010.
- [8] I. Kaya, "A genetic algorithm approach to determine the sample size for attribute control charts," *Information Sciences*, vol. 179, no. 10, pp. 1552–1566, 2009.
- [9] S. Kumar and C. S. P. Rao, "Application of ant colony, genetic algorithm and data mining-based techniques for scheduling," *Robotics and Computer-Integrated Manufacturing*, vol. 25, no. 6, pp. 901–908, 2009.
- [10] M. Z. Jahromi and M. Taheri, "A proposed method for learning rule weights in fuzzy rule-based classification systems," *Fuzzy Sets and Systems*, vol. 159, no. 4, pp. 449–459, 2008.
- [11] A. Jamali, A. Hajiloo, and N. Nariman-zadeh, "Reliability-based robust Pareto design of linear state feedback controllers using a multi-objective uniform-diversity genetic algorithm (MUGA)," *Expert Systems with Applications*, vol. 37, no. 1, pp. 401–413, 2010.
- [12] K. Suga, S. Kato, and K. Hiyama, "Structural analysis of Pareto-optimal solution sets for multi-objective optimization: An application to outer window design problems using Multiple Objective Genetic Algorithms," *Building and Environment*, vol. 45, no. 5, pp. 1144–1152, 2010.
- [13] N. Orlic and S. Loncaric, "Earthquake-explosion discrimination using genetic algorithm-based boosting approach," *Computers and Geosciences*, vol. 36, no. 2, pp. 179–185, 2010.
- [14] C.-W. Tsai, C.-H. Huang, and C.-L. Lin, "Structure-specified IIR filter and control design using real structured genetic algorithm," *Applied Soft Computing Journal*, vol. 9, no. 4, pp. 1285–1295, 2009.
- [15] L. Araujo, H. Zaragoza, J. R. Pérez-Agüera, and J. Pérez-Iglesias, "Structure of morphologically expanded queries: a genetic algorithm approach," *Data and Knowledge Engineering*, vol. 69, no. 3, pp. 279–289, 2010.
- [16] C. Bogani, M. G. Gasparo, and A. Papini, "Generalized pattern search methods for a class of nonsmooth optimization problems with structure," *Journal of Computational and Applied Mathematics*, vol. 229, no. 1, pp. 283–293, 2009.
- [17] X. Song, H. Gu, X. Zhang, and J. Liu, "Pattern search algorithms for nonlinear inversion of high-frequency Rayleigh-wave dispersion curves," *Computers and Geosciences*, vol. 34, no. 6, pp. 611–624, 2008.
- [18] Y. Zhang, L. Wu, Y. Huoc, and S. Wang, "A novel global optimization method- Genetic pattern search," *Applied Mechanics and Materials*, vol. 44–47, pp. 3240–3244, 2011.
- [19] L. De Giovanni and F. Pezzella, "An improved genetic algorithm for the distributed and flexible job-shop scheduling problem," *European Journal of Operational Research*, vol. 200, no. 2, pp. 395–408, 2010.
- [20] T. A. Srivier, J. W. Chrissis, and M. A. Abramson, "Pattern search ranking and selection algorithms for mixed variable simulation-based optimization," *European Journal of Operational Research*, vol. 198, no. 3, pp. 878–890, 2009.
- [21] L. D. S. Coelho, J. G. Sauer, and M. Rudek, "Differential evolution optimization combined with chaotic sequences for image contrast enhancement," *Chaos, Solitons and Fractals*, vol. 42, no. 1, pp. 522–529, 2009.
- [22] R. Ghanbari and N. Mahdavi-Amiri, "Solving bus terminal location problems using evolutionary algorithms," *Applied Soft Computing Journal*, vol. 11, no. 1, pp. 991–999, 2011.
- [23] S. Chaplot, L. M. Patnaik, and N. R. Jagannathan, "Classification of magnetic resonance brain images using wavelets as input to support vector machine and neural network," *Biomedical Signal Processing and Control*, vol. 1, no. 1, pp. 86–92, 2006.
- [24] E.-S. A. El-Dahshan, T. Hosny, and A.-B. M. Salem, "Hybrid intelligent techniques for MRI brain images classification," *Digital Signal Processing: A Review Journal*, vol. 20, no. 2, pp. 433–441, 2010.
- [25] Y. Zhang, Z. Dong, L. Wu, and S. Wang, "A hybrid method for MRI brain image classification," *Expert Systems with Applications*, vol. 38, no. 8, pp. 10049–10053, 2011.
- [26] X. Delaunay, M. Chabert, V. Charvillat, and G. Morin, "Satellite image compression by post-transforms in the wavelet domain," *Signal Processing*, vol. 90, no. 2, pp. 599–610, 2010.
- [27] J. Camacho, J. Picó, and A. Ferrer, "Data understanding with PCA: structural and variance information plots," *Chemometrics and Intelligent Laboratory Systems*, vol. 100, no. 1, pp. 48–56, 2010.
- [28] R. J. May, H. R. Maier, and G. C. Dandy, "Data splitting for artificial neural networks using SOM-based stratified sampling," *Neural Networks*, vol. 23, no. 2, pp. 283–294, 2010.
- [29] P. Kordík, J. Koutník, J. Drchal, O. Kovářík, M. Čepeck, and M. Šnorek, "Meta-learning approach to neural network optimization," *Neural Networks*, vol. 23, no. 4, pp. 568–582, 2010.
- [30] P. Barmapalexis, F. I. Kanaze, K. Kachrimanis, and E. Georagarakis, "Artificial neural networks in the optimization of a nimodipine controlled release tablet formulation," *European Journal of Pharmaceutics and Biopharmaceutics*, vol. 74, no. 2, pp. 316–323, 2010.
- [31] Z. W. Geem and W. E. Roper, "Energy demand estimation of South Korea using artificial neural network," *Energy Policy*, vol. 37, no. 10, pp. 4049–4054, 2009.

## Research Article

# Modeling and Analysis of the Weld Bead Geometry in Submerged Arc Welding by Using Adaptive Neurofuzzy Inference System

Nuri Akkas,<sup>1</sup> Durmuş Karayel,<sup>2</sup> Sinan Serdar Ozkan,<sup>2</sup> Ahmet Oğur,<sup>3</sup> and Bayram Topal<sup>4</sup>

<sup>1</sup> Faculty of Technical Education, Sakarya University, Sakarya, Turkey

<sup>2</sup> Department of Mechatronics Engineering, Faculty of Technology, Sakarya University, Sakarya, Turkey

<sup>3</sup> Department of Mechanical Engineering, Faculty of Engineering, Sakarya University, Sakarya, Turkey

<sup>4</sup> Department of Business, Faculty of Business, Sakarya University, Sakarya, Turkey

Correspondence should be addressed to Durmuş Karayel; [dkarayel@sakarya.edu.tr](mailto:dkarayel@sakarya.edu.tr)

Received 30 May 2013; Revised 29 August 2013; Accepted 13 September 2013

Academic Editor: Saeed Balochian

Copyright © 2013 Nuri Akkas et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This study is aimed at obtaining a relationship between the values defining bead geometry and the welding parameters and also to select optimum welding parameters. For this reason, an experimental study has been realized. The welding parameters such as the arc current, arc voltage, and welding speed which have the most effect on bead geometry are considered, and the other parameters are held as constant. Four, three, and five different values for the arc current, the arc voltage, and welding speed are used, respectively. So, sixty samples made of St 52-3 material were prepared. The bead geometries of the samples are analyzed, and the thickness and penetration values of the weld bead are measured. Then, the relationship between the welding parameters is modeled by using artificial neural network (ANN) and neurofuzzy system approach. Each model is checked for its adequacy by using test data which are selected from experimental results. Then, the models developed are compared with regard to accuracy. Also, the appropriate welding parameters values can be easily selected when the models improve.

## 1. Introduction

The submerged arc welding (SAW) is one of the manufacturing methods which are widely used. The mechanical properties of the welding joints are directly dependent on the geometrical form of bead and its properties. At the same time, the form of bead and its properties change according to the process parameters. Therefore, the process parameters must be selected so that an appropriate weld bead can be formed. There is no linear relationship between the welding parameters and weld bead geometry, and empirical formulas and experimental results are generally used for this relation. Most times, this case is incapable to select the optimum parameters values. Therefore, researchers have begun to use artificial intelligence technologies and statistics analysis methods in order for, the optimum parameters values to be selected and the relationship between the values defining bead geometry and welding parameters can be found.

Chandel et al. have developed software for theoretical predictions of the effect of current, electrode polarity, electrode diameter, and electrode extension on the melting rate, bead height, bead width, and weld penetration, in submerged-arc welding. They have predicted the weld bead geometry and melting rates of both the submerged and the metal arc welding processes by this software. The software is based on the algorithms developed by Yang et al. for predicting the weld bead geometry. This model predicts bead geometry for bead-on-plate (BOP) welds only. The variables required for input are current, voltage, travel speed, electrode diameter, electrode extension, and the electrode polarity [1].

Li et al. modeled the nonlinear relationship between the five geometric descriptors (height, width, penetration, fused and deposited areas) of a bead and the welding parameters (current, voltage, and welding speed) of submerged arc welding using neural networks. They have shown the advantages of single-output networks by a comparative study

between multioutput networks and single-output networks, each modeling one geometric descriptor. The structure of a conventional feed-forward multilayer perception network with a single output is modified to accommodate an offset layer which offsets the inputs. This network, known as the self-adaptive offset network (SAON), has definite advantages over conventional multilayer perception networks. Altogether, 21 single-output neural networks have been trained for the four types of SAW welds investigated [2].

Gunaraj and Murugan developed an application of response surface methodology for predicting weld bead quality in submerged arc welding of pipes. In their study, the variables for input are open-circuit voltage, wire feed rate, speed, and nozzle-to-plate distance. The variables for output are penetration, reinforcement, bead width, and dilution [3]. Gunaraj and Murugan also investigated the effect of process variables on the area of the heat-affected zone for the bead-on-plate and bead-on-joint in submerged arc welding of pipes by using response surface methodology. The effect of controllable process variables on the heat input and the area of the heat-affected zone (HAZ) for bead-on-plate and bead-on-joint welding was calculated and analyzed using mathematical models developed for the submerged arc welding of pipes [4]. Tušek developed four mathematical models for calculation of melting rate in arc fusion welding with a wire in coil form. The mathematical models permit calculation of melting rate in direct current welding with single-wire and double-wire electrodes (both polarities). For single-wire welding, the models treated have been improved with regard to the ones published in the literature; for twin-wire welding, these are the first models for calculation of melting rate. The mathematical models have already been tested in practice and the results obtained show that they are very accurate, simple, and applicable to practice [5]. Wikle III et al. used sensing technique for penetration depth control of the submerged arc welding process. They investigated the development of a rugged, low cost, and point infrared sensor to monitor. At the end of the study, they maintained constant depth of penetration using the infrared sensor in the presence of these perturbations by feedback control of the welding process parameters [6]. Murugan and Gunaraj studied prediction and control of weld bead geometry and shape relationships in submerged arc welding of pipes. They have developed mathematical models for submerged arc welding of pipes using five-level factorial techniques to predict three critical dimensions of the weld bead geometry and shape relationships. The models were checked for their adequacy and significance by using the  $F$ -test and the  $t$ -test, respectively. They have presented main and interaction effects of the process variables on bead geometry and shape factors in graphical form [7]. Tarnig et al. used the grey-based Taguchi methods to determine submerged arc welding process parameters in hardfacing. They presented a new approach. In this new approach, the grey relational analysis is adopted to solve the submerged arc welding process with multiple weld qualities. They obtained a grey relational grade from the grey relational analysis which is used as the performance characteristic in the Taguchi method [8]. Kanjilal et al. investigated combined effect of

flux and welding parameters on chemical composition and mechanical properties of submerged arc weld metal. In this study, rotatable designs based on statistical experiments for mixtures were developed to predict the combined effect of flux mixture and welding parameters on submerged arc weld metal chemical composition and mechanical properties by second-order regression model. Bead-on-plate weld deposits on low carbon steel plates were made at different flux composition and welding parameter combinations. The variables for input are current, voltage, welding speed, electrode stick-out, and polarity. The variables for output are chemical composition, yield strength, ultimate tensile strength, percent elongation, and Charpy impact toughness and hardness [9].

Nart and Celik propose a new practical approach in modeling to catch the correct shape of the weld pool. They predict temperature distributions and residual stresses for a plate using user subroutines and observe that the finite element results get closer to those of experimental measurements as mesh size gets finer. The residual stresses have been estimated well enough for irregular bead cross-sections by using the new approach for finite element modeling of arc-welding process [10]. Zhao et al. investigated V-I (voltage-current) curve as the monitoring signature to explore a real-time and *in situ* small scale resistance spot welding (SSRSW) quality monitoring method. They performed a systematic research on the V-I curve. Then, they proposed five factors extracted from the V-I curve to estimate the weld quality through an artificial intelligence algorithm. As a result, the study shows that the V-I signature could be used as a reliable [11]. Cho et al. studied the analysis of submerged arc welding process by three-dimensional computational fluid dynamics simulations. In the study, they adopt the Abel inversion method with CCD camera images for direct and alternating current polarities. Then, they validated simulated weld pool profiles with corresponding experimental results and were found to be in good agreement [12].

Acherjee et al. developed a nonlinear model to establish a correlation between the laser transmission welding parameters and output variables by applying artificial neural network (ANN). The process parameters consisted of laser power, welding speed, stand-off distance and clamping pressure, and the output parameters were lap-shear strength and weld-seam width. They used experimental data to train and test the network and then confirm the simulation data obtained from the neural network with the experimental data and, so, show that there was a good agreement between the experimental and numerical results [13]. Shen et al. aimed to determine how variation in heat input achieved was using single and double wires. For this reason, they measured specimens of submerged arc welded plates of ASTM A709 Grade 50 steel considering bead width, penetration depth, contact angle, heat affected zone (HAZ) size, deposition area, penetration area, and total molten area. Then, the level of dilution and different melting efficiencies were analyzed according to measuring results. They showed that the electrode melting efficiency increased initially and then decreased with increasing heat input, but the plate melting efficiency and percentage dilution changed only slightly with it [14]. Leitner et al. studied on the evaluation of fillet weld properties and fatigue

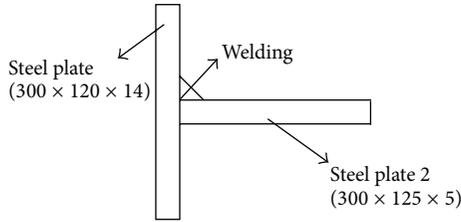


FIGURE 1: The form of welding joint.

behavior in dependence of welding parameters. They selected weld joints and investigated the influence due to the welding process parameters, especially for high-strength steels, the effect of both the geometrical, and metallurgical notch. Also, they performed the experimental fatigue tests to determine the link between fatigue life and manufacturing process-dependent weld toe notch design. Finally, they adjusted the material and manufacturing properties using the temperature profiles [15]. In this paper, Nagesh and Datta proposed an integrated method with a new approach using experimental design matrix of experimental designs technique. They explained the application of neural network for predicting the weld bead geometric descriptors and the use of genetic algorithm for optimization of process parameters. Also, they attempted to model the welding process for predicting the bead shape parameters of welded joints and used multiple linear regression techniques to develop mathematical models for weld bead shape parameters. In addition, they studied to predict the bead shape parameters using back-propagation neural network and then to optimize the process parameters for the desired front height to front width ratio and back height to back width ratio by applying genetic algorithmic approach [16].

Sathiya et al. investigated the weld bead geometry such as depth of penetration (DP), bead width (BW), and tensile strength (TS) of the laser welded butt joints made of AISI 904L super austenitic stainless steel. They used full factorial design method for the experimental design. Also, they developed artificial neural networks (ANN) program in MATLAB software to establish the relationships between the laser welding input parameters and used genetic algorithm (GA) for optimizing the process parameters and obtained optimum solutions for the three different gases. Also, they validated the optimized parameters with the experimental results [17]. Dhasa and Kumaran study the optimization of parameters of submerged arc weld using nonconventional techniques. In this study, bead-on-plate welds were carried out on mild steel plates using semiautomatic SAW machine. The input-output relationships of the process were carried out by regression analysis, and the weld bead width was minimized by this relationship. Finally, they compared the optimized values obtained from these techniques and obtained a very close relationship between them [18]. Besides these studies, another noteworthy point, in recent times, the researchers have focused their studies on artificial intelligence technologies to analyze the weld seam [19–21].

As seen from the reviewed literature, most of investigations are about bead-on-plate and bead on-joint welds.

TABLE 1: Welding parameters.

Parameters	Level 1	Level 2	Level 3	Level 4	Level 5
Voltage (Volt)	24	30	36	—	—
Current (Ampere)	200	300	400	500	—
Speed (m/min)	30	40	50	60	70

These methods give a general opinion about the effects of the welding parameters on the bead geometry. However, these effects will change if the joint type changes. Therefore, this study takes into consideration the corner weld of parts with different thickness.

## 2. Experimental Study

The experiments were conducted at Adapazari Plants of TIRSAN GROUP. The form and dimension of samples used for the experiment have been presented in Figure 1. The material is a group of structural and constructional steel (St 52-3). There are a lot of parameters which affect weld bead geometry, but this study takes into consideration some of the parameters as welding current, welding voltage, and welding speed because these three parameters are the most effective on the bead geometry.

It is considered that the values of these parameters have been selected from the applicable working ranges. The selected parameters for the welding process are given in Table 1.

The welding rod used is GEKA S2, 3.2 mm diameter, and the welding flux is LINCOLN 761. Fifty-five sets of test plates have been analyzed. The work piece used and the equipment used for the experiment are shown in Figures 2 and 3, respectively.

The length of the work piece welded is 300 mm. However, the length of sample cut from this work piece is 15 mm only, and it is taken from the best quality part of welded work piece. Then, these samples are prepared by the usual metallurgical polishing methods, and their macrostructures are photographed. These photographs are transferred to computer environment. There are thirty-one macrophotographs but some of them are shown in Figure 4, for example.

Also, cross-section of an ideal weld defining the bead geometry is presented in Figure 5. The total penetration area and welding thickness are considered as criterion for bead geometry. A measured graph surface on the computer display is prepared and it is used for measuring welding thickness and defining bead profile.

The welding thickness is directly measured from the macrophotographs. But an indirect method is followed to calculate the penetration area of weld bead. The penetration area of each part is individually considered, and they are expressed as  $A_1$  and  $A_2$ . This symbolization is shown in Figure 6.

The coordinates of some points on the limit profiles of weld beads are determined, and these profiles are expressed with polynomial equations. Then, these equations are integrated and so the penetration areas are calculated. Two areas are added and total penetration area is obtained. The

TABLE 2: Experimental results.

Test number	Welding parameters			Welding thickness (mm)	Results		
	Current (Ampere)	Voltage (Volts)	Speed (m/min)		Penetration area 1 (mm <sup>2</sup> ) (A1)	Penetration area 2 (mm <sup>2</sup> ) (A2)	Total area (mm <sup>2</sup> ) (A1 + A2)
1	200	24	30	5,020	4,344	8,966	13,310
4	200	24	60	3,670	0,514	7,295	7,808
6	200	30	30	5,310	3,205	23,014	26,219
7	200	30	40	4,270	5,261	22,911	28,172
9	200	30	60	3,740	1,727	15,874	17,601
10	200	30	70	2,990	1,551	14,425	15,977
11	200	36	30	5,440	10,607	17,471	28,079
16	300	24	30	7,470	5,877	48,120	53,996
17	300	24	40	6,790	10,450	34,606	45,055
18	300	24	50	5,900	9,487	34,615	44,102
19	300	24	60	4,040	1,282	22,742	24,024
20	300	24	70	2,910	0,920	19,943	20,863
21	300	30	30	6,570	13,365	51,048	64,412
22	300	30	40	5,28	5,014	72,173	77,187
23	300	30	50	4,880	5,431	61,203	66,633
25	300	30	70	4,200	9,497	39,230	48,727
27	300	36	40	6,150	10,853	60,583	71,436
29	300	36	60	5,060	11,509	42,691	54,199
32	400	24	40	7,980	4,735	51,131	55,866
33	400	24	50	7,370	15,228	37,557	52,785
35	400	24	70	6,550	8,285	39,848	48,133
36	400	30	30	8,380	15,061	113,806	128,867
37	400	30	40	7,100	7,859	103,215	111,074
40	400	30	70	4,790	7,432	55,217	62,649
42	400	36	40	7,390	11,439	92,705	104,144
44	400	36	60	5,780	12,174	58,368	70,542
45	400	36	70	5,110	9,906	46,871	56,777
52	500	30	40	11,980	10,291	106,652	116,943
53	500	30	50	11,420	9,440	80,999	90,439
54	500	30	60	8,540	1,112	81,206	82,318
55	500	30	70	7,220	7,815	63,483	71,298

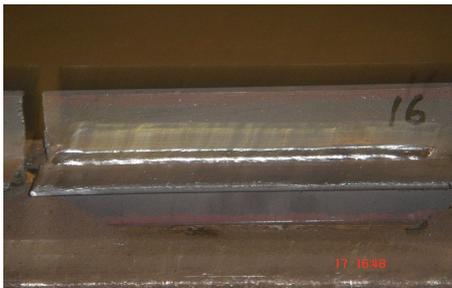


FIGURE 2: A work piece used for the experiment.



FIGURE 3: Welding machine used for the experiment.

bead geometry and the polynomial equation of the limit penetration profile of the first sample are shown in Figure 7

and in Figure 8, respectively. The experimental results are presented in Table 2.

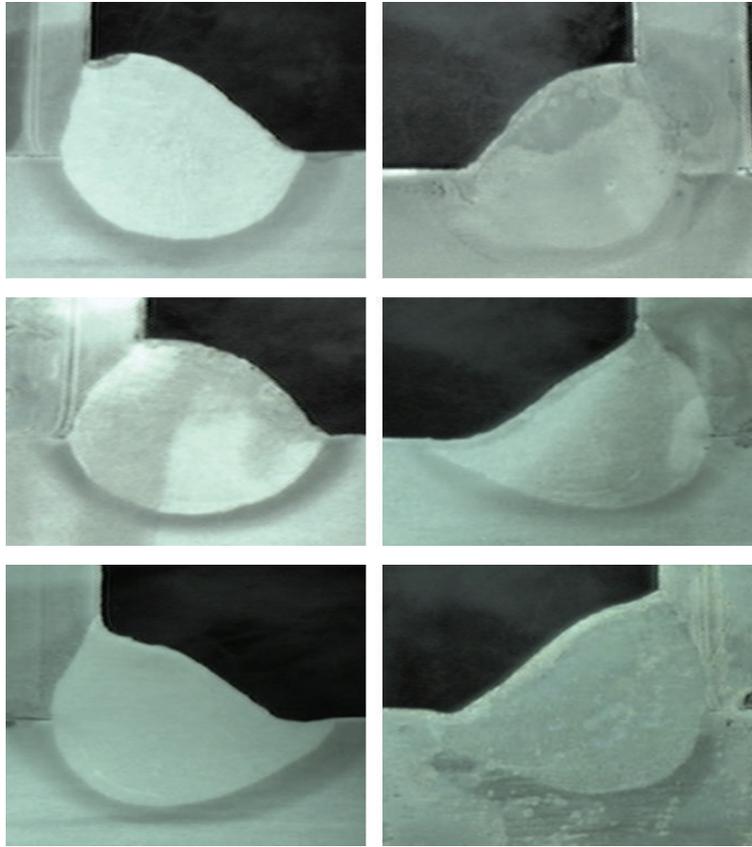


FIGURE 4: Some examples of macrophotographs of weld bead.

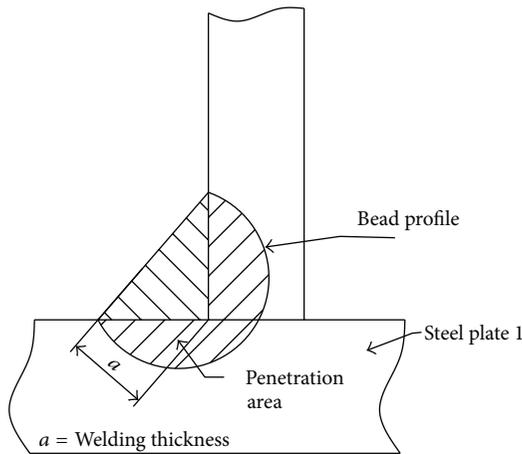
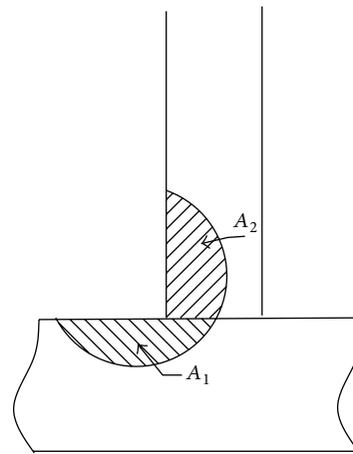


FIGURE 5: Cross-section of an ideal weld bead.



$A_1$  = The penetration area on the steel plate 1  
 $A_2$  = The penetration area on the steel plate 2

FIGURE 6: The penetration areas of the parts.

### 3. Modelling and Numerical Analyses

The weld quality depends on the weld bead geometry too much. To know correct machine setting that ensures satisfactory weld quality for a welding process is difficult. Because there is no known linear relationships between the desired bead geometry and the welding parameters. In other words, a good welding quality can be obtained if the bead geometry can be controlled by the process parameters.

This case requires establishing a mathematical model of the relationship between the bead geometry and the welding parameters. Today, artificial intelligence technologies give this possibility. In this study, artificial neural network (ANN) and neurofuzzy approach are used and they are compared.

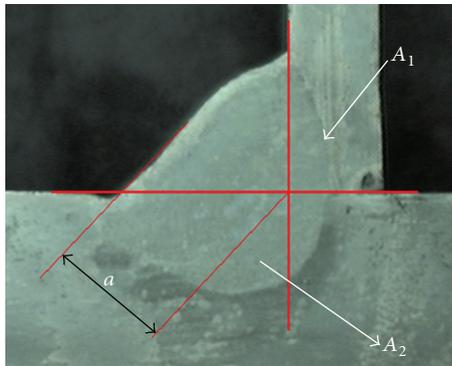


FIGURE 7: The bead geometry.

**3.1. Artificial Neural Network (ANN).** In this part of study, neural network model of submerged arc weld is established by using Neural Network Toolbox of MATLAB package. The input data consist of arc current, arc voltage, and welding speed. The output variables are welding thickness and penetration area. These output variables are individually considered and are modeled. Twenty-one data for training set and five data for testing set are used. The performance of training set for the welding thickness and the performance of testing set are shown in Figures 9 and 10, respectively.

For the penetration area, the performance of training set and the performance of testing set are shown in Figures 11 and 12, respectively.

**3.2. Neurofuzzy Approach.** Neurofuzzy systems combine the positive attributes of neural networks and fuzzy systems. Adaptive neuro fuzzy inference system (ANFIS) is used to the modeling of the SAW. There are three input parameters and two output values. In this study, the output values are considered individually, and so the models are prepared as three inputs and one output. The architecture of the ANFIS used in the proposed neurofuzzy approach is shown in Figure 13.

The same ANFIS model structure is used for the welding thickness and the penetration area. However, the different ANFIS editors are employed for the welding thickness and the penetration areas. The ANFIS editor used for the welding thickness is shown in Figure 14.

The ANFIS editor enables loading data, generating Fuzzy Inference System (FIS), training FIS, and testing FIS. Twenty-six experimental data of thirty-one welding process experiments listed in Table 2 are utilized to train the used ANFIS model, and five of them are utilized to test the model. The training and testing performance of the model are shown in Figure 14 and in Figure 15, respectively. The ANFIS model for the penetration area is developed by using similar procedure and the same input data have been used for this phase. Finally, the training performance and the checking performance for penetration area have been obtained in Figure 16 and in Figure 17, respectively.

## 4. Results and Discussion

Both ANN and ANFIS have given very suitable results. This case can be seen from the checking performance diagrams.

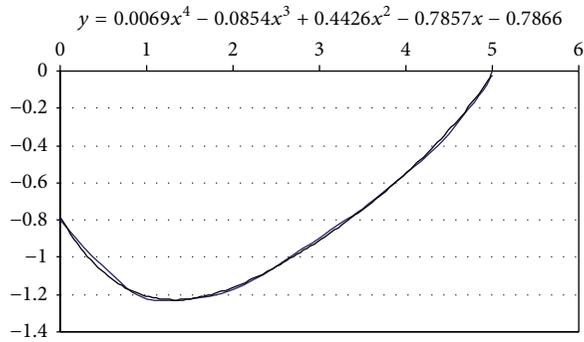
From the comparison of the results obtained, it can be observed that the results of ANFIS are closer than the results of ANN to the experimental results. Therefore, to present the results of ANFIS is preferred and the effects of the welding thickness and penetration area are individually shown in Figures 18(a) and 18(b) and in Figures 19(a), 19(b), and 19(c), respectively. The input and the output values are reduced by using certain proportions. Therefore, the results of ANN and ANFIS must be extended with the same factors. This factor is 1000 for the arc current and the penetration area. For the arc voltage, the welding speed, and the welding thickness it is 100.

Experimental and theoretic results show that arc current, arc voltage, and welding speed affect penetration area and welding thickness in SAW process. Figure 18(a) shows the effect of interaction between arc voltage and arc current on welding thickness. It can be observed that the welding thickness increases with a decrease in arc voltage and an increase in arc current. But the maximum welding thickness is obtained when arc voltage is of minimum value and arc current is of maximum value. Also, the minimum arc current and the maximum arc voltage correspond to the minimum welding thickness. Figure 18(b) presents the effect of interaction between arc voltage and welding speed on welding thickness. It can be seen that the welding thickness increases with a decrease in arc voltage and in welding speed. The minimum welding thickness corresponds to the maximum values of arc voltage and arc current. However, the maximum welding thickness occurs when welding speed and arc voltage are minimum value.

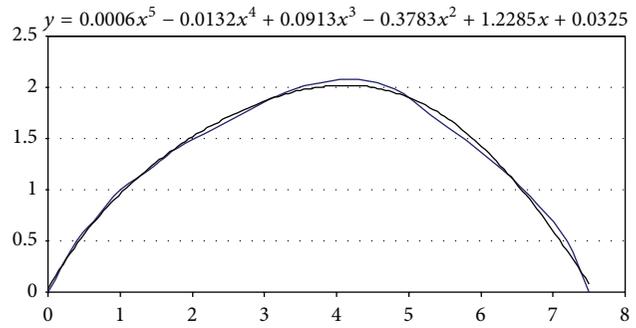
Figures 19(a), 19(b), and 19(c) present the effects of the welding parameters on bead penetration area. Figure 19(a) shows that the penetration area is almost stationary up to 0,31 arc voltage values, and it has a tendency to increase when arc current is 0,35 and arc voltage is higher than 0,31. The minimum penetration area occurs when voltage is maximum value and arc current is of minimum value. As seen from Figure 19(b), the penetration area increases with an increase in arc voltage and welding speed and it reaches the maximum value when welding speed and voltage are the maximum values. The minimum value of penetration area occurs when welding speed is of minimum value and arc voltage is 0,3. Figure 19(c) shows that the penetration area increases with an increase in arc voltage and welding speed, and it reaches the minimum value where arc current and welding speed are minimum values. On the contrary, the penetration area is the maximum value where arc current and welding speed are the maximum values.

## 5. Conclusions

In this study, it is aimed to obtain a relationship between the values defining bead geometry and the welding parameters and to select optimum welding parameters. For this reason, an experimental study has been realized. Also, modeling and analysis of the weld bead geometry in submerged arc welding by using adaptive neurofuzzy inference system have been performed. The major conclusions drawn from this study are the following.



(a) Macro 1: Area 1 (A1).



(b) Macro 1: Area 2 (A2).

FIGURE 8: The polynomial equations representing the limit penetration profile of weld bead.

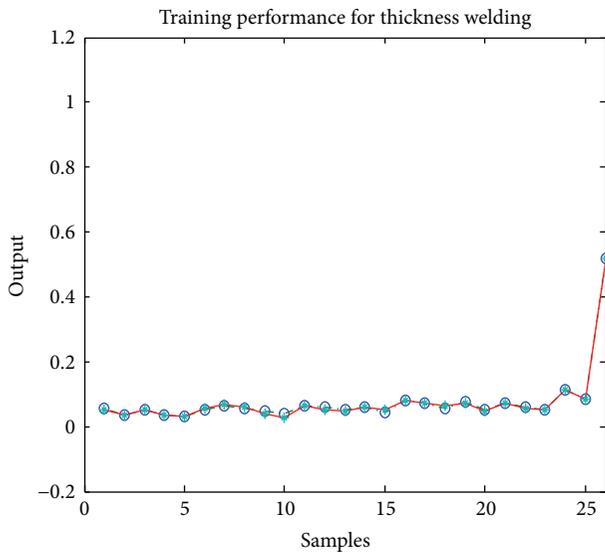


FIGURE 9: The performance of training set for the welding thickness.

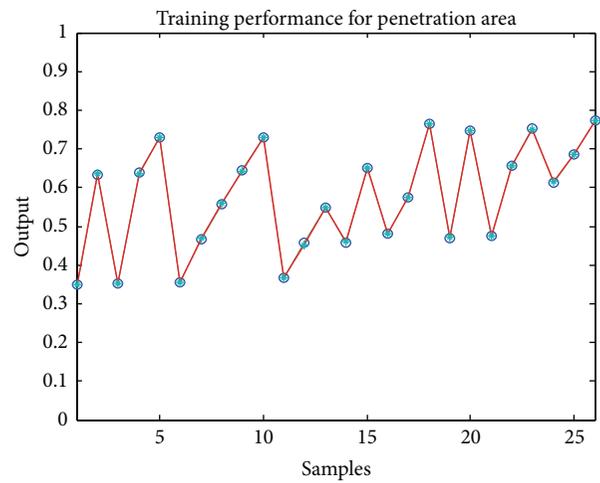


FIGURE 11: The performance of training set for the penetration area.

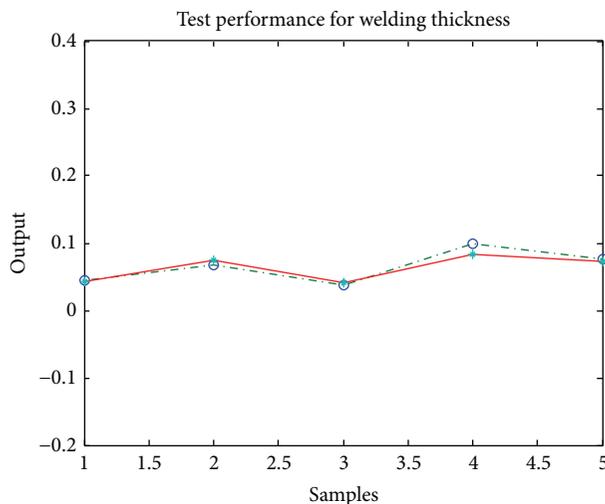


FIGURE 10: The performance of test set for the welding thickness.

- (i) ANN and ANFIS results are very close each other. But the results of ANFIS are closer than the results of ANN to the experimental results.
- (ii) The experimental and theoretical results show that arc current, arc voltage, and welding speed affect penetration area and welding thickness in SAW process.
- (iii) It can be observed that the welding thickness increases with a decrease in arc voltage and an increase in arc current. But the maximum welding thickness is obtained when arc voltage is of minimum value and arc current is of maximum value.
- (iv) The minimum arc current and the maximum arc voltage correspond to the minimum welding thickness.
- (v) The welding thickness increases with a decrease in arc voltage and in welding speed.
- (vi) The minimum welding thickness corresponds to the maximum values of arc voltage and arc current.
- (vii) The maximum welding thickness occurs when welding speed and arc voltage are of minimum value.

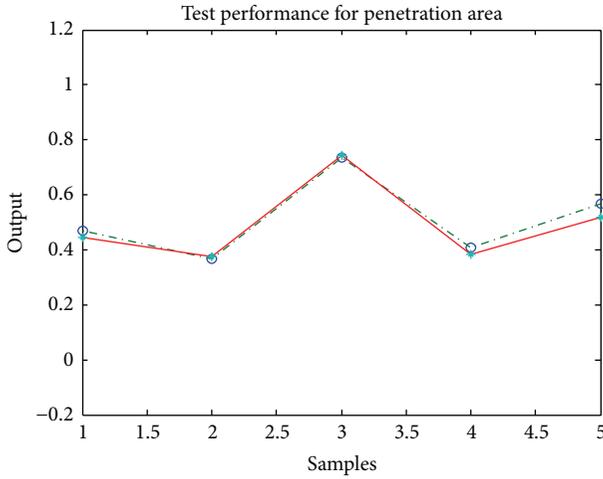


FIGURE 12: The performance of testing set for the penetration area.

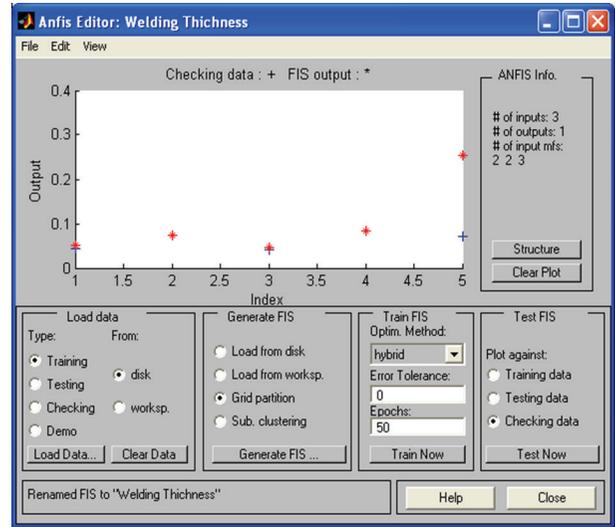


FIGURE 15: The checking performance for the welding thickness.

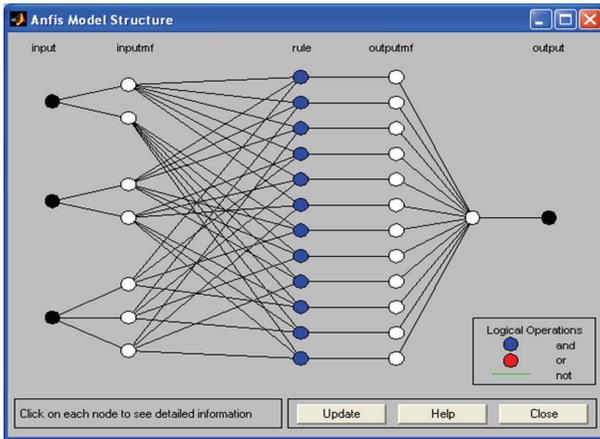


FIGURE 13: The architecture of the ANFIS used in the proposed approach.

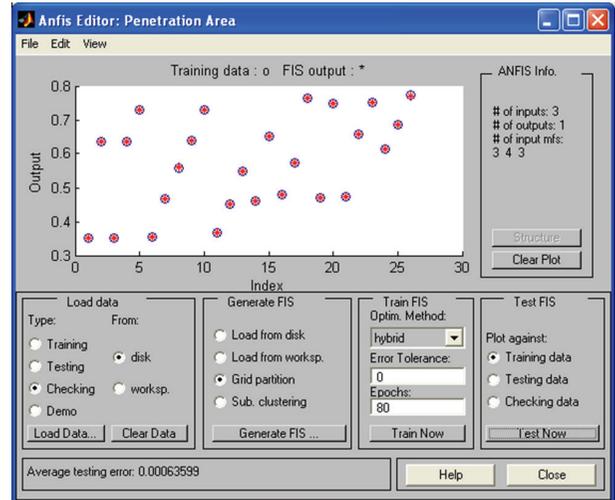


FIGURE 16: The training performance for penetration area.

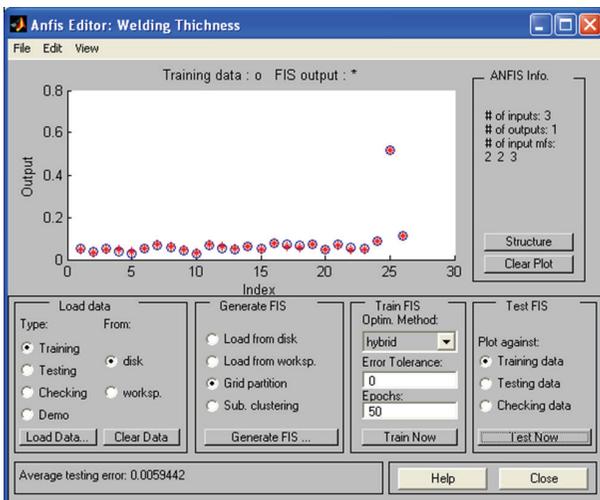


FIGURE 14: The training performance for the welding thickness.

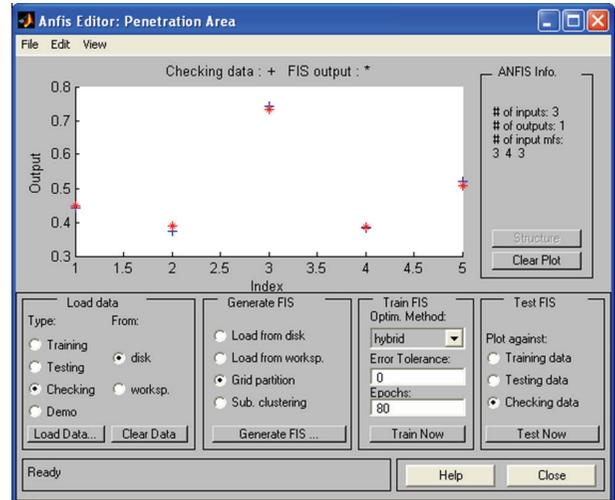


FIGURE 17: The checking performance for penetration area.

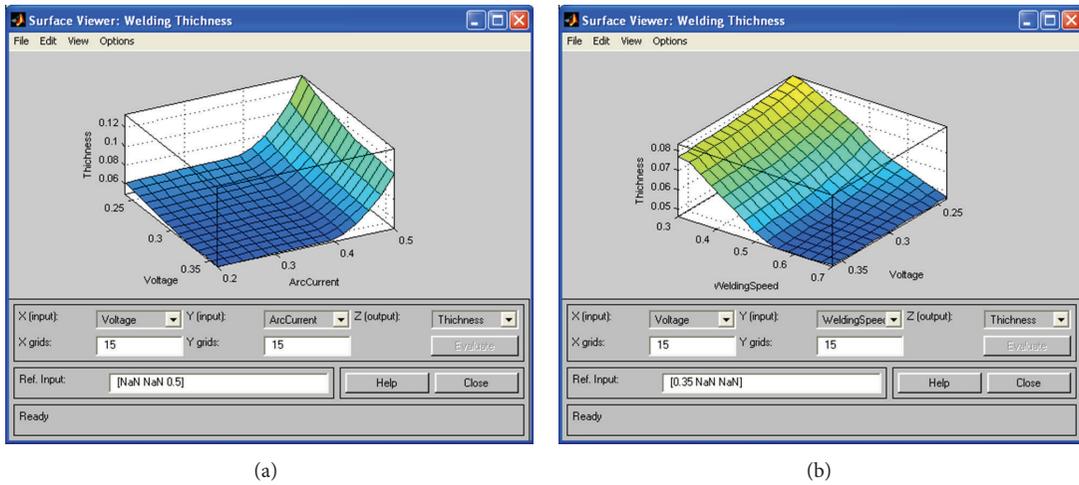


FIGURE 18: (a) The effect of arc voltage and arc current on welding thickness. (b) The effect of welding speed and arc voltage on welding thickness.

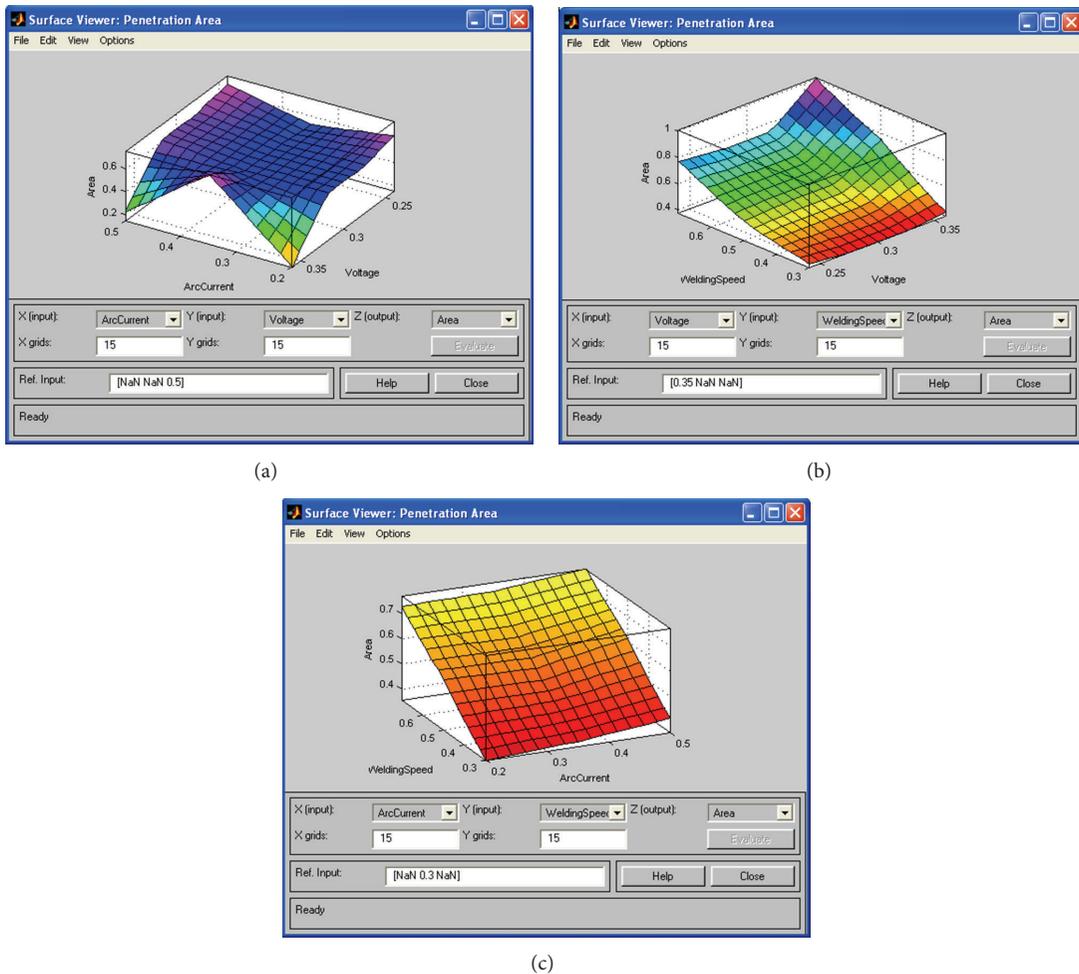


FIGURE 19: (a) The effect of arc voltage and arc current on penetration area. (b) The effect of arc voltage and welding speed on penetration area. (c) The effect of arc current and welding speed on penetration area.

- (viii) The penetration area is almost stationary up to 0,31 are voltage values, and it has a tendency to increase when arc current is 0,35 and arc voltage is higher than 0,31.
- (ix) The minimum penetration area occurs when voltage is of maximum value and arc current is of minimum value.
- (x) The penetration area increases with an increase in arc voltage and welding speed, and it reaches the maximum value when welding speed and voltage are the maximum values.
- (xi) The minimum value of penetration area occurs when welding speed is of minimum value and arc voltage is 0,3.
- (xii) The penetration area increases with an increase in arc voltage and welding speed, and it reaches the minimum value where arc current and welding speed are minimum values. On the contrary, the penetration area is the maximum value where arc current and welding speed are the maximum value.

It has been shown that submerged arc welding (SAW) process can be modeled by using artificial intelligence technologies. Finally, the models developed are able to predict the welding parameters required to obtain the desired bead geometry. This study can help to develop an intelligence control system for SAW process.

## References

- [1] R. S. Chandel, H. P. Seow, and F. L. Cheong, "Effect of increasing deposition rate on the bead geometry of submerged arc welds," *Journal of Materials Processing Technology*, vol. 72, no. 1, pp. 124–128, 1997.
- [2] P. Li, M. T. C. Fang, and J. Lucas, "Modelling of submerged arc weld beads using self-adaptive offset neural networks," *Journal of Materials Processing Technology*, vol. 71, no. 2, pp. 288–298, 1997.
- [3] V. Gunaraj and N. Murugan, "Application of response surface methodology for predicting weld bead quality in submerged arc welding of pipes," *Journal of Materials Processing Technology*, vol. 88, no. 1, pp. 266–275, 1999.
- [4] V. Gunaraj and N. Murugan, "Prediction and comparison of the area of the heat-affected zone for the bead-on-plate and bead-on-joint in submerged arc welding of pipes," *Journal of Materials Processing Technology*, vol. 95, no. 1–3, pp. 246–261, 1999.
- [5] J. Tušek, "Mathematical modeling of melting rate in twin-wire welding," *Journal of Materials Processing Technology*, vol. 100, no. 1, pp. 250–256, 2000.
- [6] H. C. Wikle III, S. Kottilingam, R. H. Zee, and B. A. Chin, "Infrared sensing techniques for penetration depth control of the submerged arc welding process," *Journal of Materials Processing Technology*, vol. 113, no. 1–3, pp. 228–233, 2001.
- [7] N. Murugan and V. Gunaraj, "Prediction and control of weld bead geometry and shape relationships in submerged arc welding of pipes," *Journal of Materials Processing Technology*, vol. 168, no. 3, pp. 478–487, 2005.
- [8] Y. S. Tarn, S. C. Juang, and C. H. Chang, "The use of grey-based Taguchi methods to determine submerged arc welding process parameters in hardfacing," *Journal of Materials Processing Technology*, vol. 128, no. 1–3, pp. 1–6, 2002.
- [9] P. Kanjilal, T. K. Pal, and S. K. Majumdar, "Combined effect of flux and welding parameters on chemical composition and mechanical properties of submerged arc weld metal," *Journal of Materials Processing Technology*, vol. 171, no. 2, pp. 223–231, 2006.
- [10] E. Nart and Y. Celik, "A practical approach for simulating submerged arc welding process using FE method," *Journal of Constructional Steel Research*, vol. 84, pp. 62–71, 2013.
- [11] D. Zhao, Y. Wanga, Z. Lin, and S. Sheng, "An effective quality assessment method for small scale resistance spot welding based on process parameters," *NDT & E International*, vol. 55, pp. 36–41, 2013.
- [12] D. W. Cho, W. H. Song, M. H. Cho, and S. J. Na, "Analysis of submerged arc welding process by three-dimensional computational fluid dynamics simulations," *Journal of Materials Processing Technology*, vol. 213, pp. 2278–2291, 2013.
- [13] B. Acherjee, S. Mondal, B. Tudu, and D. Misra, "Application of artificial neural network for predicting weld quality in laser transmission welding of thermoplastics," *Applied Soft Computing Journal*, vol. 11, no. 2, pp. 2548–2555, 2011.
- [14] S. Shen, I. N. A. Oguocha, and S. Yannacopoulos, "Effect of heat input on weld bead geometry of submerged arc welded ASTM A709 Grade 50 steel joints," *Journal of Materials Processing Technology*, vol. 212, no. 1, pp. 286–294, 2012.
- [15] M. Leitner, T. Fössl, M. Stoschka, and W. Eichlseder, "Evaluation of fillet weld properties and fatigue behaviour in dependence of welding parameters," *Archives of Civil and Mechanical Engineering*, vol. 11, no. 3, pp. 651–660, 2011.
- [16] D. S. Nagesh and G. L. Datta, "Genetic algorithm for optimization of welding variables for height to width ratio and application of ANN for prediction of bead geometry for TIG welding process," *Applied Soft Computing Journal*, vol. 10, no. 3, pp. 897–907, 2010.
- [17] P. Sathiy, K. Panneerselvam, and R. Soundararajan, "Optimal design for laser beam butt welding process parameter using artificial neural networks and genetic algorithm for super austenitic stainless steel," *Optics and Laser Technology*, vol. 44, no. 6, pp. 1905–1914, 2012.
- [18] J. E. R. Dhas and S. Kumanan, "Optimization of parameters of submerged arc weld using non conventional techniques," *Applied Soft Computing Journal*, vol. 11, no. 8, pp. 5198–5204, 2011.
- [19] R. Sudhakaran, V. VeL Murugan, K. M. Senthil Kumar et al., "Effect of welding process parameters on weld bead geometry and optimization of process parameters to maximize depth to width ratio for stainless steel gas tungsten arc welded plates using genetic algorithm," *European Journal of Scientific Research*, vol. 62, no. 1, pp. 76–94, 2011.
- [20] J. E. R. Dhas and S. Kumanan, "Optimization of parameters of submerged arc weld using non conventional techniques," *Applied Soft Computing Journal*, vol. 11, no. 8, pp. 5198–5204, 2011.
- [21] P. Sreeraj and T. Kannan, "Modelling and prediction of Stainless steel clad bead geometry deposited by GMAW using regression and artificial neural network models," *Advances in Mechanical Engineering*, vol. 2012, Article ID 237379, 12 pages, 2012.

## Research Article

# Nonlinear Predictive Control of Mass Moment Aerospace Vehicles Based on Ant Colony Genetic Algorithm Optimization

**Xiaoyu Zhang, Peng Li, Dexin Xu, Ben Mao, and Kunpeng He**

*College of Automation, Harbin Engineering University, Harbin 150001, China*

Correspondence should be addressed to Xiaoyu Zhang; [zhangxiaoyu@hrbeu.edu.cn](mailto:zhangxiaoyu@hrbeu.edu.cn)

Received 26 April 2013; Revised 5 August 2013; Accepted 30 August 2013

Academic Editor: Vishal Bhatnaga

Copyright © 2013 Xiaoyu Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Based on the mathematical model of the mass moment aerospace vehicles (MMAV), a coupled nonlinear dynamical system is established by rational simplification. The flight control system of MMAV is designed via utilizing nonlinear predictive control (NPC) approach. Aiming at the parameters of NPC is generally used the trial-and-error method to optimize and design, a novel kind of NPC parameters optimization strategy based on ant colony genetic algorithm (ACGA) is proposed in this paper. The method for setting NPC parameters with ACA in which the routes of ants are optimized by the genetic algorithm (GA) is derived. And then, a detailed realized process of this method is also presented. Furthermore, this optimization algorithm of the NPC parameters is applied to the flight control system of MMAV. The simulation results show that the system not only meets the demands of time-response specifications but also has excellent robustness.

## 1. Introduction

The aerodynamic rudder control is generally used for the traditional flight missile in the atmosphere. But in upper air, especially in the exoatmospheric space above 30 kilometers, because of lower air density, the dynamic pressure becomes very low. In the far field of missile fighting, because the speed of missile reduces, the dynamic pressure is relatively low. This causes the dynamic characteristics of the missile to be decreased and the maneuverability to be weak. The weak maneuverability of the missile makes the missile miss the targets frequently and cannot meet the demands of precise guidance and control. This requests the new technical approach. In order to enhance the maneuverability, agility, and guidance precision of the anti-aircraft missile, a new method of mass moment control is proposed in 2000 by USA. From the related literatures, the moving-mass-center control was studied by both Russia and USA. In the 1990s of the 20th century. But, because the maneuver control technology of the reentry warhead was kept secret extremely in the world, the public literatures and the related research are very few. The related research which can be seen in the literature has almost focused on reentry warhead at

high Mach numbers. Several studies have suggested that mass moment control system (MMCS) appears to offer the greater design and cost advantages [1–8]. There are some advantages of mass moment control as follows. (1) All the mechanism of MMCS is in aerospace vehicles, which will not affect the aerodynamic configuration and is better to accuracy of terminal attack. (2) The actuators of MMCS are internal moving masses, which decrease the thermal load of aerodynamic configuration and avoid the gap on the surface of the vehicle and ablation steering surface. (3) By using aerodynamic forces generated by high-speed flight of vehicle, we can decrease the energy consumption and get effective control avoiding conflict between fuel consumption and control moment generated by lateral jet engine.

The stability control mode of the three channels was used for MMAV in this paper. The two orthogonal moving masses were arranged on radial of MMAV, and another moving mass was arranged on axis. Each of the channels has strong coupled aerodynamic, control, inertia, and dynamics. So MMAV becomes one multivariable system model of the nonlinearity and uncertainty. A coupled nonlinear dynamic model is obtained by the rational simplification for this

system. The nonlinear predictive control (NPC) of MMAV is investigated in the paper. By considering the insufficiency of NPC adjustment parameters, one parameter optimization method is proposed based on ant colony genetic algorithm (ACGA). With the merits of GA, this method has the global search ability, latent parallel computing, and simple computation process and can solve the contradiction of the optimal solution and the searching space. The contradiction of searching space and time performance is also overcome, which easy to convergent in nonglobal optimal solution and computing time too long when using ant colony algorithm (ACA) to solve optimization question. This method is applied to system design of flight control on MMAV.

During the last decade, much effort has been made in GA to investigate aerospace vehicles systems [9–18]. For instance, [9] demonstrates the advantages of using a real coded genetic algorithm (GA) for aerospace engineering design applications. In [10], in order to realize the high precision attitude determination of the space crafts by star sensors, an optimization method of star map distorted model based on improved genetic algorithm is presented. Reference [12] presented a new method in hierarchical genetic algorithms (HGAs) to speed up the optimization of aerodynamic shapes. In [13], an improved GA for airfoil shape parameterization is presented which takes into consideration the characteristics of viscous transonic flow particularly around the trailing edge. In [14], a novel parsimonious genetic programming (PGP) algorithm together with a novel aeroengine optimum data-driven dynamic start process model based on PGP is proposed. In order to obtain the global solution efficiently, [15] proposed a hybrid optimizer combining the advantages of a floating-coded genetic algorithm and simplex method. In [16], an optimization process, based on a genetic algorithm, is used to meet the frequency domain handling qualities requirements for the longitudinal plane. It is well known that GA cannot make good use of system feedback information. It often results in a lot of redundancy and reduces the computing efficiency. In contrast, ant colony algorithm accumulates and renews the optimal path information continuously. It has the ability of parallel processing and global searching. Many scholars have researched the hybrid algorithm including GA and ACA and got some good results in applications. For example, [19] gave an application of ant colony, genetic algorithm, and data mining-based techniques for scheduling. [20] used the hybrid algorithm in the contrast of images. Reference [21] addressed the optimization of cloud database route scheduling based on combination of GA and ACA. More results can be found from [22–28]. However, to the best of our knowledge, there are few works appeared to investigate the MMCS by the GA combined with ACA. This motivates the research in this paper.

This paper is organized as follows. The mathematical model of MMAV is derived in Section 2. In Section 3, nonlinear predictive control and its robustness analysis are employed to design MMCS. Ant colony genetic algorithm is applied to parameter optimization of MMCS in Section 4. In Section 5, simulation demonstrates the ability of the optimized controller to effectively control the MMAV's motion. Conclusion is given in Section 6.

## 2. The Mathematical Model of MMAV

**2.1. The General Dynamics Model of the MMAV.** The basic principle by which MMCS is able to control the vehicle's motion is to produce the control torque by using the aerodynamic forces and moving the masses within the MMAV to offset the c. m. of system.

Suppose that the MMAV includes  $n$  moving masses and the mass of MMAV's shell is  $m_B$ . The mass of the  $i$ th moving mass is  $m_i$ . So, the total mass of MMAV is  $m_C = m_B + \sum_{i=1}^n m_i$ . The mass ratio of the  $i$ th moving mass is  $\mu_i = m_i/m_C$ . The coordinates in the body fixed frame are  $\delta_i = (x_{1m}, y_{1m}, z_{1m})^T$ ,  $i = 1, 2, \dots, n$ . In the ground frame, the velocity of the center of MMAV is  $V_{AB} = \dot{\delta}_{AB}$ , and the acceleration is  $\dot{V}_{AB} = \ddot{\delta}_{AB}$ . Let the coordinates in the ground with the body fixed frame of the  $i$ th moving mass be  $\delta_{di}$  and the coordinates in the ground frame  $\delta_{Ai}$ . There is the relationship  $\delta_{di} = L_{A1}\delta_{1i}$ ,  $\delta_{Ai} = \delta_{di} + \delta_{AB}$ .

The coordinates of MMAV's c. m. in the ground frame are given by

$$\begin{aligned}\delta_{AC} &= \frac{1}{m_C} \left( m_B \delta_{AB} + \sum_{i=1}^n m_i \delta_{Ai} \right) \\ &= \delta_{AB} + \sum_{i=1}^n \mu_i L_{A1} \delta_{1i} \\ &= \delta_{AB} + L_{A1} \sum_{i=1}^n \mu_i \delta_{1i}.\end{aligned}\quad (1)$$

After derivation, the translational equation of the MMAV in the ground frame can be presented below:

$$\begin{aligned}\dot{V}_{AB} &= \frac{R}{m_C} + g - \left( (\omega_1)_A^{\times*} + (\omega_1)_A^{\times 2} \right) \\ &\quad \times L_{A1} \sum_{i=1}^n \mu_i \delta_{1i} - 2(\omega_1)_A^{\times} L_{A1} \\ &\quad \times \sum_{i=1}^n \mu_i \dot{\delta}_{1i} - L_{A1} \sum_{i=1}^n \mu_i \ddot{\delta}_{1i}.\end{aligned}\quad (2)$$

Then, the translational equation of the MMAV in the body fixed frame is as follows:

$$\begin{aligned}\dot{V}_{1B} + (\omega_1)_1^{\times} V_{1B} &= \frac{R_1}{m_C} + g_1 \\ &\quad - \left( (\omega_1)_1^{\times*} + (\omega_1)_1^{\times 2} \right) \sum_{i=1}^n \mu_i \delta_{1i} \\ &\quad - 2(\omega_1)_1^{\times} \sum_{i=1}^n \mu_i \dot{\delta}_{1i} - \sum_{i=1}^n \mu_i \ddot{\delta}_{1i},\end{aligned}\quad (3)$$

where  $L_{A1}$  is the transformation matrix from the body fixed frame to the ground frame and  $(\omega_1)_A^{\times}$  is the antisymmetric matrix of the angular velocity of the MMAV in the ground frame.  $(\omega_1)_1^{\times}$  is the antisymmetric matrix of the angular velocity of the MMAV in the body fixed frame.

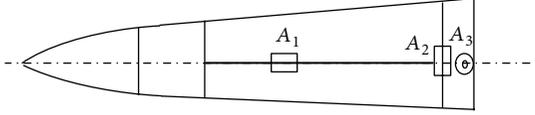


FIGURE 1: . MMAV with three moving masses.

Correspondingly, the force equation of the  $i$ th moving mass in the body fixed frame is given by

$$\begin{aligned}
 F_{1i} = & \mu_i R_1 + m_i \left[ \left( (\omega_1)_1^{\times*} + (\omega_1)_1^{\times 2} \right) \delta_{1i} \right. \\
 & \left. + 2(\omega_1)_1^{\times} \dot{\delta}_{1i} + \ddot{\delta}_{1i} \right] \\
 & - m_i \left[ \left( (\omega_1)_1^{\times*} + (\omega_1)_1^{\times 2} \right) \right. \\
 & \left. \times \sum_{i=1}^n \mu_i \delta_{1i} + 2(\omega_1)_1^{\times} \right. \\
 & \left. \times \sum_{i=1}^n \mu_i \dot{\delta}_{1i} + \sum_{i=1}^n \mu_i \ddot{\delta}_{1i} \right]. \quad (4)
 \end{aligned}$$

According to D'Alembert principle, the rotational equation in body coordinates is obtained as follows:

$$\left( \frac{dH}{dt} \right)_1 + M_{1F} = \frac{dH_1}{dt} + (\omega_1)_1^{\times} H_1 + M_{1F} = M_{1R}, \quad (5)$$

where  $M_{1F} = -\sum_{i=1}^n S_{1i} F_{1i}$ ,  $S_{1i}$  is the antisymmetric matrix of the  $i$ th moving mass in the body fixed frame representing the position coordinates  $\delta_{1i}$ .

**2.2. The Dynamics Model of the MMAV with Three Moving Masses.** The structure diagram of the MMAV with three moving masses is shown in Figure 1. To quickly adjust the flying attitude and decrease coupling, one mass is fixed at  $x$ -axis in the body fixed frame. Other two masses are fixed at  $y$ -axis and  $z$ -axis in radial direction through the MMAV's axis.

The mass of the MMAV's shell is  $m_B$ . The mass of axial moving mass  $A_1$  is  $m_1$ , and the coordinate in the body fixed frame is  $\delta_{11}$ . The mass of radial moving mass  $A_2$  is  $m_2$ , and the coordinate in the body fixed frame is  $\delta_{12}$ . The mass of radial moving mass  $A_3$  is  $m_3$ , and the coordinate in the body fixed frame is  $\delta_{13}$ . So, the total mass is  $m_C = m_B + m_1 + m_2 + m_3$ , and the mass ratios are  $\mu_1 = m_1/m_C$ ,  $\mu_2 = m_2/m_C$ , and  $\mu_3 = m_3/m_C$ , respectively.

This section derives the equations of motion fully accounted for the dynamic coupling between the four bodies. The moving masses are allowed to translate with respect to the MMAV's shell but are not allowed to rotate with respect to the MMAV's shell. Both the MMAV and the moving masses are assumed to be rigid bodies.

In the body fixed frame, the interaction between axial moving mass  $A_1$  and MMAV is  $F_{11}$ , the interaction between radial moving mass  $A_2$  and MMAV is  $F_{12}$ , and the interaction

between radial moving mass  $A_3$  and MMAV is  $F_{13}$ . Equation (4) can be presented below:

$$\begin{aligned}
 F_{11} = & m_1 \left[ \dot{V}_{1B} + (\omega_1)_1^{\times} V_{1B} \right. \\
 & \left. + \left( (\omega_1)_1^{\times*} + (\omega_1)_1^{\times 2} \right) \delta_{11} \right. \\
 & \left. + 2(\omega_1)_1^{\times} \dot{\delta}_{11} + \ddot{\delta}_{11} - g_1 \right], \\
 F_{12} = & m_2 \left[ \dot{V}_{1B} + (\omega_1)_1^{\times} V_{1B} \right. \\
 & \left. + \left( (\omega_1)_1^{\times*} + (\omega_1)_1^{\times 2} \right) \delta_{12} \right. \\
 & \left. + 2(\omega_1)_1^{\times} \dot{\delta}_{12} + \ddot{\delta}_{12} - g_1 \right], \quad (6) \\
 F_{13} = & m_3 \left[ \dot{V}_{1B} + (\omega_1)_1^{\times} V_{1B} \right. \\
 & \left. + \left( (\omega_1)_1^{\times*} + (\omega_1)_1^{\times 2} \right) \delta_{13} \right. \\
 & \left. + 2(\omega_1)_1^{\times} \dot{\delta}_{13} + \ddot{\delta}_{13} - g_1 \right].
 \end{aligned}$$

The vector translational dynamics of MMAV can be obtained by (3) as follows:

$$\begin{aligned}
 & \dot{V}_{1B} + (\omega_1)_1^{\times} V_{1B} + (\omega_1)_1^{\times*} (\mu_1 \delta_{11} + \mu_2 \delta_{12} + \mu_3 \delta_{13}) \\
 = & \frac{R_1}{m_C} + g_1 - \left[ (\omega_1)_1^{\times 2} (\mu_1 \delta_{11} + \mu_2 \delta_{12} + \mu_3 \delta_{13}) \right. \\
 & \left. + 2(\omega_1)_1^{\times} (\mu_1 \dot{\delta}_{11} + \mu_2 \dot{\delta}_{12} + \mu_3 \dot{\delta}_{13}) \right. \\
 & \left. + (\mu_1 \ddot{\delta}_{11} + \mu_2 \ddot{\delta}_{12} + \mu_3 \ddot{\delta}_{13}) \right]. \quad (7)
 \end{aligned}$$

The rotational dynamics of MMAV obtained by (5) are given by

$$\begin{aligned}
 & \left[ J_1 + (1 - \mu_1) m_1 S_{11}^T S_{11} + (1 - \mu_2) m_2 S_{12}^T S_{12} \right. \\
 & \left. + (1 - \mu_3) m_3 S_{13}^T S_{13} - \mu_2 m_1 S_{11} S_{12}^T \right. \\
 & \left. - \mu_3 m_1 S_{11} S_{13}^T - \mu_1 m_2 S_{12} S_{11}^T - \mu_3 m_2 S_{12} S_{13}^T \right. \\
 & \left. - \mu_1 m_3 S_{13} S_{11}^T - \mu_2 m_3 S_{13} S_{12}^T \right] (\omega_1)_1^* \\
 = & M_{1R} - (\mu_1 S_{11} + \mu_2 S_{12} + \mu_3 S_{13}) R_1 \\
 & - (\omega_1)_1^{\times} J_1 (\omega_1)_1 \\
 & + [\mu_1 m_2 S_{12} + \mu_1 m_3 S_{13} - (1 - \mu_1) m_1 S_{11}]
 \end{aligned}$$

$$\begin{aligned}
& \times \left( (\omega_1)_1^{\times 2} \delta_{11} + 2(\omega_1)_1^{\times} \delta_{11} + \ddot{\delta}_{11} \right) \\
& + [\mu_2 m_1 S_{11} + \mu_2 m_3 S_{13} - (1 - \mu_2) m_2 S_{12}] \\
& \times \left( (\omega_1)_1^{\times 2} \delta_{12} + 2(\omega_1)_1^{\times} \delta_{12} + \ddot{\delta}_{12} \right) \\
& + [\mu_3 m_1 S_{11} + \mu_3 m_2 S_{12} - (1 - \mu_3) m_3 S_{13}] \\
& \times \left( (\omega_1)_1^{\times 2} \delta_{13} + 2(\omega_1)_1^{\times} \delta_{13} + \ddot{\delta}_{13} \right), \tag{8}
\end{aligned}$$

where the antisymmetric matrixes of position coordinates of the three moving masses in the body fixed frame are  $S_{11}$ ,  $S_{12}$ , and  $S_{13}$ , respectively.

Furthermore, the equations of motion of MMAV system also include some relative movement functions and nonlinear aerodynamic functions. The equations of motion clearly indicate that the MMCS is a complex nonlinear system which has the variable coefficients and large disturbances caused by the accelerations and velocities of masses.

### 3. Nonlinear Predictive Control and Robustness Analysis

*3.1. Nonlinear Predictive Control.* Consider the following MMAV nonlinear system [29]:

$$\begin{aligned}
\dot{x} &= f(x) + g(x)u \\
y &= h(x), \tag{9}
\end{aligned}$$

where  $[\omega_{x1} \ \omega_{y1} \ \omega_{z1} \ \gamma \ n_y \ n_z]^T = [x_1 \ x_2 \ x_3 \ x_4 \ x_5 \ x_6]^T$  is the state variable,  $u = [\delta_x \ \delta_y \ \delta_z]^T \in R^3$  is the control input, and  $y = [x_4 \ x_5 \ x_6]^T$  is the output to be controlled by the control input  $u$ .

Reference output signals of the system are

$$y_r(t) = [y_{1r}(t) \ y_{2r}(t) \ y_{3r}(t)]^T. \tag{10}$$

Generally,  $y_{1r}(t)$ ,  $y_{2r}(t)$ , and  $y_{3r}(t)$  are chosen to be step signal.

Then the output tracking errors of the system are

$$\begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} = \begin{bmatrix} y_1 - y_{1r}(t) \\ y_2 - y_{2r}(t) \\ y_3 - y_{3r}(t) \end{bmatrix}. \tag{11}$$

According to the mathematical model forms of MMAV, select function  $S(x)$  is as follows:

$$S(x) = \begin{bmatrix} \dot{e}_1 + c_1 e_1 + c_2 x_{S1} \\ \dot{e}_2 + c_3 e_2 + c_4 x_{S2} \\ \dot{e}_3 + c_5 e_3 + c_6 x_{S3} \end{bmatrix}, \tag{12}$$

where  $\dot{x}_{S1} = e_1$ ,  $\dot{x}_{S2} = e_2$ , and  $\dot{x}_{S3} = e_3$ .

So,  $S(x)$  can be as the linear combination of the output tracking error, its differential term, and its integral term.

Furthermore the choice from  $c_1$  to  $c_6$  should meet the following conditions:

$$\begin{aligned}
\lambda^2 + c_1 \lambda + c_2 &= (\lambda + \lambda_1)(\lambda + \lambda_2) \\
\lambda^2 + c_3 \lambda + c_4 &= (\lambda + \lambda_3)(\lambda + \lambda_4) \\
\lambda^2 + c_5 \lambda + c_6 &= (\lambda + \lambda_5)(\lambda + \lambda_6), \tag{13}
\end{aligned}$$

where  $\lambda_i > 0$ ,  $i = 1, 2, 3, 4, 5, 6$ .

Derivation of function  $S(x)$  is

$$\dot{S}(x) = A(x) + D(x)u. \tag{14}$$

In order to obtain the predictive control law, the quadratic objective function is chosen as follows:

$$J(u) = \frac{1}{2} \left[ S^T(t+h)QS(t+h) + u^T(t)Ru(t) \right]. \tag{15}$$

$Q$  is a positive definite symmetric matrix,  $R$  is a positive semidefinite symmetric matrix,  $h$  is a small positive number, and  $S(t+h)$  is the predictive value of  $S$  at time  $t+h$ .  $S(t+h)$  is expanded approximately at  $t$  according to the first-order Taylor's series:

$$S(t+h) \approx S(t) + h\dot{S}(t) = S(t) + h(A + Du). \tag{16}$$

In order to make  $J(u)$  least, set  $\partial J/\partial u = 0$  in (15). The control law is obtained below:

$$u = -h(R + h^2 D^T Q D)^{-1} D^T Q (S + hA), \tag{17}$$

when  $R = 0$ , and  $D^{-1}$  exist, substitute formula (17) into formula (16) and obtain formula (18) below:

$$\dot{S}(x) + \frac{S(x)}{h} = 0. \tag{18}$$

Obviously, when  $t \rightarrow \infty$ ,  $S(x)$  tends to zero. When  $R \neq 0$  and  $D^{-1}$  exist,  $S(x)$  also tends to be boundary by simulation. Because  $S(x)$  includes integral term of tracking error, the output tracking error tends to zero inevitably.

*3.2. Robustness Analysis of the System.* Consider the uncertain nonlinear system

$$\begin{aligned}
\dot{x} &= \hat{f}(x) + \Delta f(x) + [\hat{g}(x) + \Delta g(x)]u \\
y &= h(x), \tag{19}
\end{aligned}$$

where  $\hat{f}(x)$  and  $\hat{g}(x)$  are the parts of the nominal system and  $\Delta f(x)$  and  $\Delta g(x)$  are the uncertainty parts of the system.

Suppose that  $\hat{f}(x)$  and  $\hat{g}(x)$  are the known bounded continuous functions and  $\Delta f(x)$  and  $\Delta g(x)$  satisfy the following conditions:

$$\begin{aligned}
\|\Delta f(x)\| &< \|\hat{f}(x)\| \\
\|\Delta g(x)\| &< \|\hat{g}(x)\|. \tag{20}
\end{aligned}$$

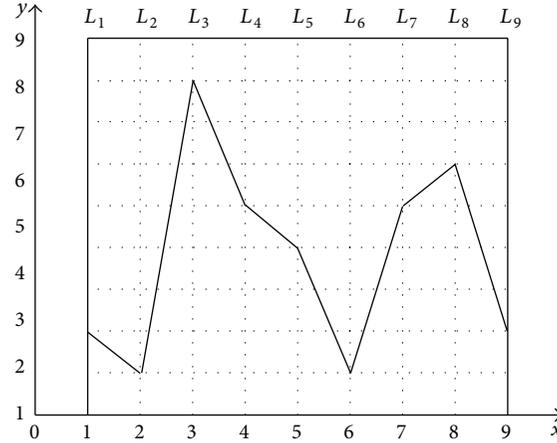


FIGURE 2: Schematic plot on the generation of nodes and routes.

Then, each term in (17) can be written as the following form:

$$\begin{aligned} A(x) &= \widehat{A}(x) + \Delta A(x) \\ D(x) &= \widehat{D}(x) + \Delta D(x) \\ S(x) &= \widehat{S}(x) + \Delta S(x), \end{aligned} \quad (21)$$

where  $\widehat{A}(x)$ ,  $\widehat{D}(x)$ , and  $\widehat{S}(x)$  correspond to the known nominal system and  $\Delta A(x)$ ,  $\Delta D(x)$ , and  $\Delta S(x)$  correspond to the uncertain parts of system.

Firstly, consider the situation is  $R = 0$ . According to the parameter of nominal system by formula (17), the control law is obtained by

$$u = -h^{-1} \widehat{D}^{-1} (\widehat{S} + h\widehat{A}). \quad (22)$$

Substitute formula (21) and formula (22) into formula (14):

$$\begin{aligned} \dot{S} &= -h^{-1} (1 + \Delta D \widehat{D}^{-1}) S \\ &\quad - h^{-1} (1 + \Delta D \widehat{D}^{-1}) \Delta S + \Delta A \\ &= -h^{-1} (1 + \Delta D \widehat{D}^{-1}) (S - \Delta S) + \Delta A. \end{aligned} \quad (23)$$

Because of  $\|\Delta D \widehat{D}^{-1}\| < 1$  and  $\|\Delta S\| < \|S\|$ ,  $S$  also is boundary which should be known.

Set  $S(x)$  as input and  $e$  as output. The transfer functions are obtained by (12):

$$G(s) = \begin{bmatrix} \frac{s}{s^2 + c_1 s + c_2} \\ \frac{s}{s^2 + c_3 s + c_4} \\ \frac{s}{s^2 + c_5 s + c_6} \end{bmatrix}. \quad (24)$$

From formula (24),  $G(s)$  is strict stability convergence,  $S(x)$  is boundary input, and  $e$  is boundary output inevitably.

Because  $S(x)$  includes the integral term of  $e$ ,  $e(\infty)$  should tend to zero ultimately.

To sum up, when  $R$  is equal to 0, the system has strong robustness. Similarly, when  $R \neq 0$ , the system still has strong robustness by analysis above.

## 4. Ant Colony Genetic Algorithm

**4.1. Optimization Principle of Ant Colony Algorithm.** When ants in the nature seek foods or meet obstacles, they always can find one optimal route from their nests to food source or round the obstacles. The reason is that the ants can release one special secretion called pheromone on the roads which they pass through (pheromone, this material can volatilize gradually along with time); then the following ants may choose next route according to the pheromone which the fore ants leaved. The pheromone in one route is higher, and the number of times which should be selected is more. That is, the performance of this route will be superior, and the probability that the following ants choose this route will be larger. So, a positive feedback process of learning information is established by this way. At last, the optimal solution can be obtained [30–32]. The ant colony algorithm is based on this principle.

For formula (13), when control system is designed by nonlinear predictive control, the parameters to be optimized are  $c_i > 0$ ,  $i = 1, 2, 3, 4, 5, 6$ . In the generality, set  $c_1 = c_3 = c_5$  and  $c_2 = c_4 = c_6$ . So, only  $c_1$  and  $c_2$  need to be optimized.

**4.2. Generation of Nodes and Routes.** By the inspiration of [33], the two parameters to be optimized are expressed in a plane for using the ant colony algorithm expediently. The graph is shown in Figure 2.

$L_1 \sim L_5$  and  $L_6 \sim L_9$  express  $c_1$  and  $c_2$ , respectively. According to the test results in [7], the initial value  $c_1$  is 200.00 and the initial value  $c_2$  is 10.00.  $c_1$  and  $c_2$  are calculated, respectively, by the formulas below:

$$\begin{aligned} c_1 &= y_{1,j} \times 10^2 + y_{2,j} \times 10^1 + y_{3,j} \\ &\quad \times 10^0 + y_{4,j} \times 10^{-1} + y_{5,j} \times 10^{-2} \end{aligned} \quad (25)$$

$$c_2 = y_{6,j} \times 10^1 + y_{7,j} \times 10^0 + y_{8,j} \times 10^{-1} + y_{9,j} \times 10^{-2}. \quad (26)$$

Take the fold line in Figure 2 as example. The expressed parameters are  $c_1 = 218.54$  and  $c_2 = 15.62$ .

**4.3. Establishment of the Objective Function.** When the objective function is established, the dynamic performances of the system on establishing overload instructions should be made as basis according to system requests for design, which mainly consider system overshoot, the rise time, and the settling time. The objective function is defined as follows:

$$F = \lambda_\delta \left( \frac{\delta}{\delta_0} \right) + \lambda_{tr} \left( \frac{t_r}{t_{r0}} \right) + \lambda_{ts} \left( \frac{t_s}{t_{s0}} \right), \quad (27)$$

where  $\delta_0$ ,  $t_{r0}$ , and  $t_{s0}$  are the overshoot, the rise time, and the settling time which guidance control system requested.  $\lambda_\delta$ ,  $\lambda_{tr}$ , and  $\lambda_{ts}$  are the weight coefficients, respectively. Their values are 0.4, 0.2, and 0.4.

**4.4. The Choice of the Route Nodes.** Suppose that it takes each ant the equal time to crawl from any node on line  $L_i$  to the next any node on line  $L_{i+1}$  and has nothing to do with the distance between the nodes. Therefore, if all ants start from the origin of coordinates, they will simultaneously arrive at each line  $L_i$  ( $i = 1 \sim 9$ ). Finally, they will also simultaneously arrive at their end points on the line  $L_9$ . Then, one circulation is completed. Define  $\tau(x_i, y_{i,j}, t)$  to express the remaining pheromone on node  $(x_i, y_{i,j})$  at  $t$ . The pheromone is equal on each node in the initial time. In order to make the algorithm search more feasible solutions in the preliminary stage,  $\tau(x_i, y_{i,j}) = \tau_{\max}$  and  $\Delta\tau(x_i, y_{i,j}, 0) = 0$  are set at initialization. Suppose that  $P_k(x_i, y_{i,j}, t)$  expresses the probability in which the ant  $k$  crawls from any node on  $L_{i-1}$  to node  $(x_i, y_{i,j})$  at  $t$ ; then

$$P_k(x_i, y_{i,j}, t) = \frac{\tau^\alpha(x_i, y_{i,j}, t) \eta^\beta(x_i, y_{i,j}, t)}{\sum_{j=0}^9 \tau^\alpha(x_i, y_{i,j}, t) \eta^\beta(x_i, y_{i,j}, t)}, \quad (28)$$

in which  $\eta(x_i, y_{i,j}, t)$  is the visibility on node  $(x_i, y_{i,j})$ ,

$$\eta(x_i, y_{i,j}, t) = \frac{10 - |y_{i,j} - y_{i,j}^*|}{10}, \quad (29)$$

where the value  $y_{i,j}^*$  is obtained as the following method. In the first circulation of ant colony algorithm,  $y_{i,j}^*$  is given by the  $y$ -coordinate value of each node which  $c_1$  and  $c_2$  map in Figure 2. In each circulation later,  $y_{i,j}^*$  is given by the  $y$ -coordinate value of each node in which  $c_1$  and  $c_2$  correspond to the optimal route produced in the previous circulation (the optimal performances in previous circulation) map in Figure 2.

**4.5. Renewal and Limitation of the Pheromone.** Suppose that  $t = 0$  at initial time; all ants are located at the origin of coordinates  $O$ . After 9 units of time, all ants crawl from the

initial point to the end point. The pheromone in the route may be adjusted by the following formula:

$$\tau(x_i, y_{i,j}, t + 9) = \rho(x_i, y_{i,j}, t) + \Delta\tau(x_i, y_{i,j}). \quad (30)$$

For using more optimal solution information during the running period of algorithm, the renewal and limitary mechanism of the pheromone can use the maximal minimal ant system (MMAS), in which only the one optimal ant can increase pheromone after each iterative. And there is  $\tau(x_i, y_{i,j}) \in [\tau_{\min}, \tau_{\max}]$ :

$$\Delta\tau_{(x_i, y_{i,j})}^{\text{best}} = \frac{q}{F_{\text{best}}(t)}. \quad (31)$$

$F_{\text{best}}(t)$  represents the optimal objective function value in this circulation. It can be calculated by (27).

**4.6. The Steps of Optimizing MMCS Parameter Using ACGA.** The hybridization of AC and GA has been studied in many works. The methods have been successfully applied to diverse combinatorial optimization problems including traveling salesman, quadratic assignment, vehicle routing, telecommunication networks, image processing, constraint satisfaction, and scheduling. Recently, researchers have been dealing with the relation of ACGA to the fields of optimal control and reinforcement learning, in which, with the merits of GA, this method has the global search ability, latent parallel computing, and simple computation process and can solve the contradiction of the optimal solution and the searching space. It also can overcome the contradiction of searching space and time performance, in contrast with easing to converge to nonglobal optimal solution and computing time too long when using AC to solve optimization question. So, the control parameters optimization method of MMAV is proposed based on ACGA. The goals are to meet the demands of time-response specifications and improve the robustness for MMCS. Several studies have suggested that MMCS appears to offer the greater design and cost advantages in flight vehicles control. We present the mathematical modeling process of MMAV and describe the method of nonlinear predictive control for MMCS. But it is very difficult to design the MMCS because there are many parameters to adjust manually. In this paper, a new method is proposed using ACGA, by which automatic parameter optimization can be realized instead of manually adjusting.

When the MMCS parameters are optimized by the ant colony algorithm, the variation operation of GA can be used for the second optimization of the obtained result [19, 34]. That is, to avoid to fall into partial optimal by using ant colony algorithm to obtain a group partial solution that makes the objective function expressed by formula (27) is minimal. Then, the coding is carried on (the binary system is used in this paper), and the variation of certain digits is caused stochastically. If the variation solution surpasses the original solution in genetic and variation process, then the variation solution is retained. Otherwise, the original solution is used. After this variation, seeking in the partial minimal region can be jumped out, and the quality of solutions can be

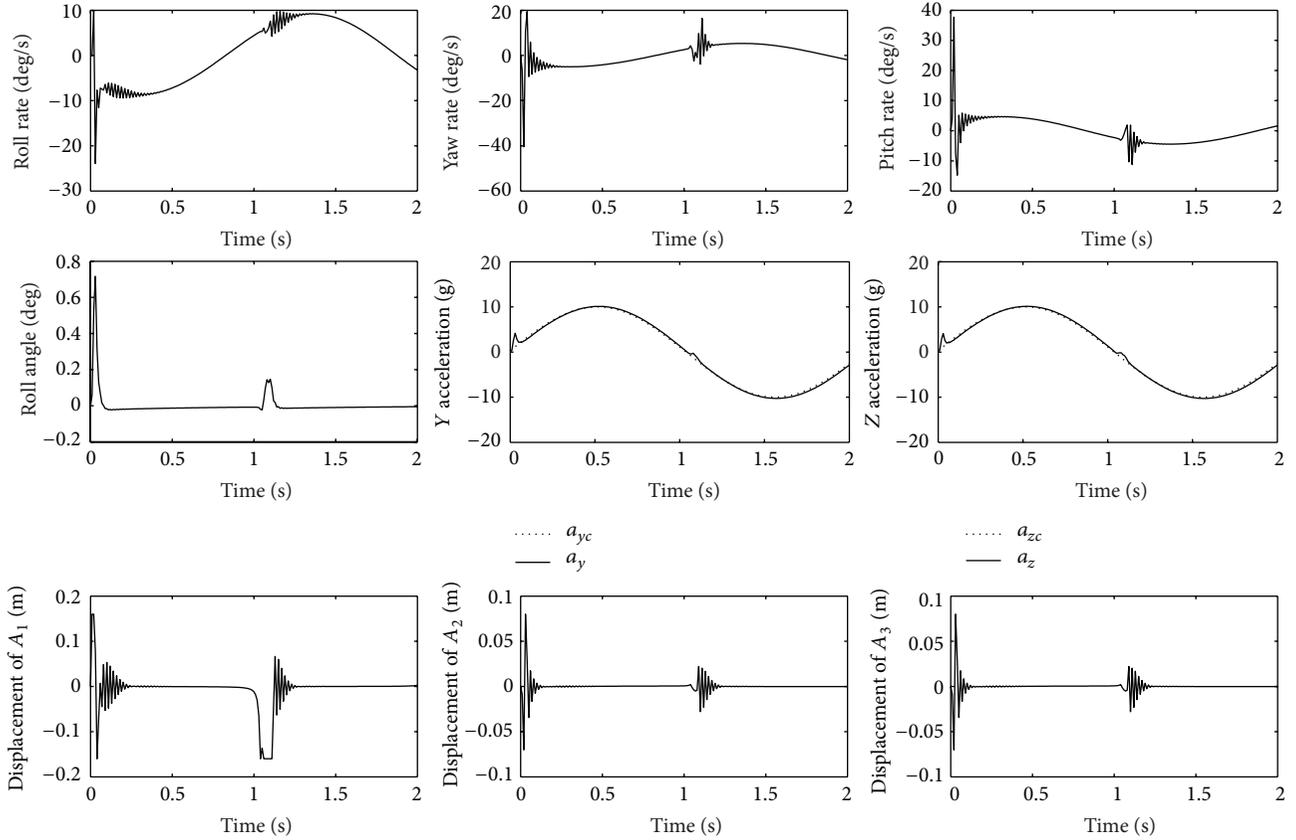


FIGURE 3: State response of MMAV using NPC based on GA optimization.

enhanced; the convergence velocity of the optimal solution is speeded up. So, ACGA can realize the highly effective, fast, and complete seeking to the search space.

The basic steps to optimize MMCS parameters with ACGA are described as follows:

*Step 1.* Initialize the MMCS parameters and the correlation ACGA parameters. The initial values of  $c_1$  and  $c_2$  can be seen above.

*Step 2.* Set ant number as  $m$ . Define the one-dimensional array  $Route_k$  including 9 elements for each ant. The  $y$ -coordinate values of 9 nodes which the ant will pass through are saved in this array, that is, the route of the ant crawling.

*Step 3.* Make time counter  $t = 0$  and cycle number  $N_c = 0$ . Define the biggest cycle number and the pheromone values on each node at initial time; set  $\Delta\tau(x_i, y_{i,j}) = 0$ . Put all the ants at the initial station  $O$ .

*Step 4.* Set  $i = 1$ .

*Step 5.* Compute the transfer probabilities that the ant moves to each node on the line  $L_i$  by formula (28). According to these probabilities, a node for each ant on the line  $L_i$  is chosen by roulette principle, and the ant also will be moved to this

node. Simultaneously, the  $y$ -coordinate value of this node is saved in  $i$ th element of  $Route_k$ .

*Step 6.* Set  $i = i + 1$ . If there is  $i \leq 9$ , jump to Step 4. Otherwise, jump to Step 7.

*Step 7.*  $c_1^k$  and  $c_2^k$  corresponding to this route are calculated by formula (25) and formula (26) according to  $Route_k$ , that is, the routes passed by the ants. The dynamic performances  $\delta^k$ ,  $t_r^k$ , and  $t_s^k$  can be calculated by carrying on computer simulation applying these parameters to the MMCS system of MMAV. Then, the objective function  $F_k$  corresponding to the ant  $k$  is calculated by formula (27), and  $F_{\text{best}}(t)$  is obtained. The optimal solution is saved in this circulation. The MMCS parameters corresponding to these also are saved into  $c_1^*$  and  $c_2^*$ . Then, the second optimization is carried on by GA. The optimal solution from the original solution and the mutation solution is chosen.

*Step 8.* Set  $t \leftarrow t + 9$  and  $N_c \leftarrow N_c + 1$ . According to formula (30) and formula (31), the pheromones on each node are renewed, and  $Route_k$  is reset to zero.

*Step 9.* If  $N_c < N_{c_{\text{max}}}$  and the whole ant colony is not convergent to one route, all ants will be put at the initial station  $O$  again, and then jump to Step 4. If  $N_c < N_{c_{\text{max}}}$ , but the whole ant colony converges to one same route, the

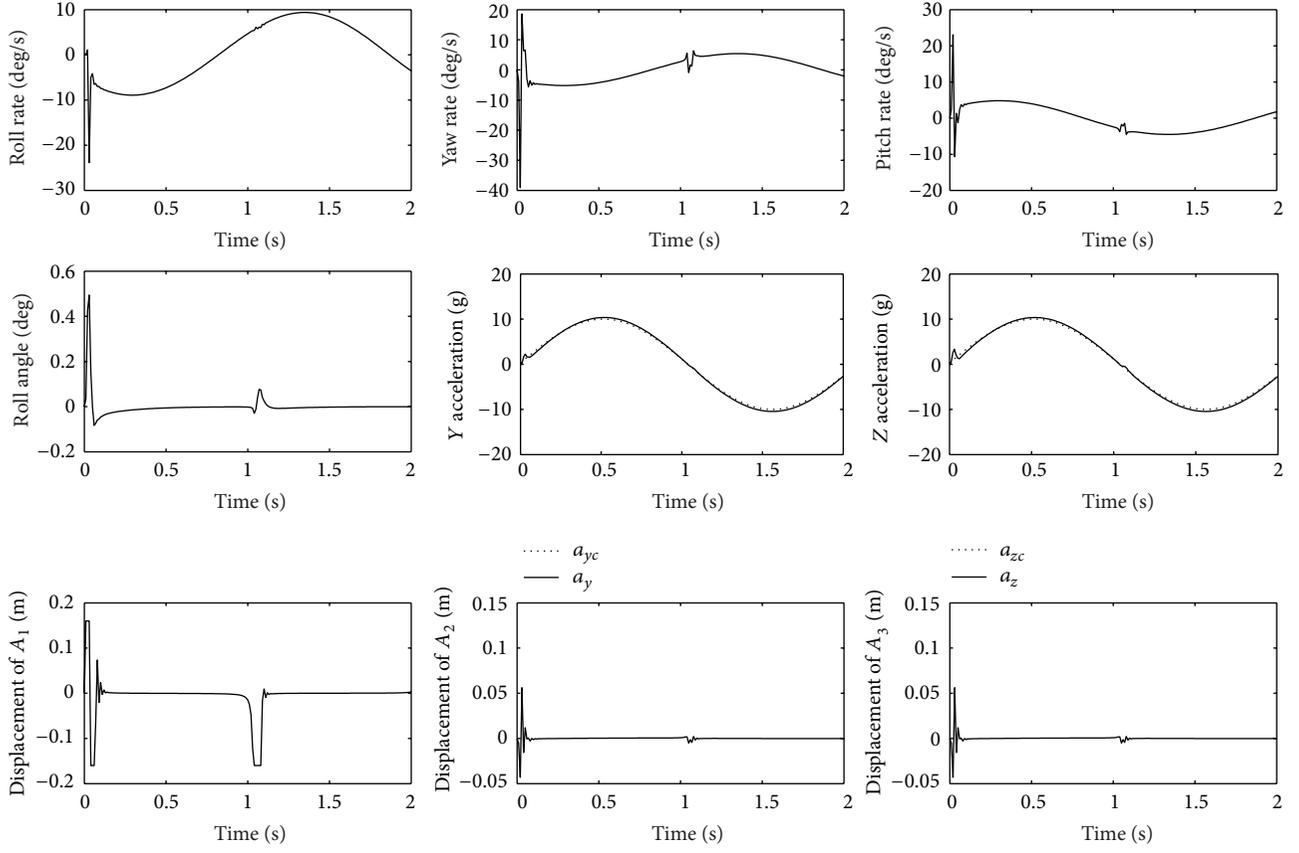


FIGURE 4: State response of MMAV using NPC based on ACGA optimization.

algorithm will be ended and the corresponding optimized MMCS parameters  $c_1^*$  and  $c_2^*$  are achieved.

## 5. Simulation Result and Discussion

Because the control performances of MMAV need to be realized through an axial moving mass and two-radial-moving-mass to and fro movement on the guide rail, the control effects of the system directly are affected by the characteristics of servo system. So, in the design of MMCS, the dynamic characteristics of servo system have to be considered.

Considering the band width of the measure elements for system (accelerometer and rate gyro) is relatively large, the simplification model of the servo system is obtained by repeated test:

$$G_R(s) = \frac{1}{T_R^2 s^2 + 2\zeta_R T_R s + 1}, \quad (32)$$

where  $T_R = 0.01$  and  $\zeta_R = 0.8$  to the axial moving mass,  $T_R = 0.005$  and  $\zeta_R = 0.8$  to the radial moving mass. Simultaneously, the saturated constraints of the displacements should be considered for the whole design assigns when the system is tested.

According to the ant colony genetic algorithm proposed in this paper to carry on parameter optimization, the final parameters of MMAV in nonlinear predictive control system are shown in Figure 2. Simultaneously, 6DOF mathematical

simulation of MMAV was carried on. The signals  $\dot{y}_{ir}$  and  $\ddot{y}_{ir}$  ( $i = 1, 2, 3$ ) in the flight control system are obtained by the second-order filter. The simulation results in which the aerodynamic parameters have 50% uncertainty are shown at length of article limited and the system design results in nominal state as the foundation. The simulation graphs are shown in Figures 3 and 4, which are state responses of MMAV using NPC with the original GA and ACGA, respectively. The time response and system stability of ACGA are better than those of the original GA. The simulation curves of the normal overload and each control of MMAV are only shown in this paper (because the radial moving mass is symmetry, the displacement curves of  $oy_1$  axis and  $oz_1$  axis for the moving mass are the same). (Only the displacement curve of  $oy_1$  axis for the moving mass is given). From the graphs, the displacement of the axial moving mass is limited in the scope  $\pm 0.16$  m, and the displacements of the two radial moving masses are limited in the scope  $\pm 0.08$  m. That flight control of MMAV may be realized by only very short displacement of the moving mass. The dynamic performances of control system for MMAV still satisfy the design requests although aerodynamic parameters have been changed 50% with their nominal values.

As shown in simulation results, the nonlinear predictive control is proposed to design MMCS, in which controller parameters are optimized with the original GA and ACGA. The simulation results show that MMCS with ACGA has

better dynamic behavior and strong robustness than that with the original GA. This approach can improve the robustness of MMCS and make it work effectively. The theoretical analysis and the simulation results of the flight control system for MMAV show that the strategy of the NPC parameter optimization based on the ant colony genetic algorithm proposed in this paper is effective and feasible. The algorithm of the flight control system for MMAV designed by using NPC based on ACGA optimization is simple. Moreover, the dynamic performances of the system satisfy the design requests completely and also have strong robustness. Simultaneously, the time is saved. The optimization algorithm proposed in this paper does not rely on the precise mathematical model of MMAV and makes processing question more flexible, adaptable, and robust. This method can not only improve the quality of control system design but also reduce the design difficulty and may solve optimization problems of the control system parameters which are designed by other control methods. So, it has important reference value in the control engineering domain.

## 6. Conclusion

An autopilot for a nonlinear six-degree-of-freedom MMAV is introduced in this paper. Because the stability control system used by MMAV with three moving masses is three-channel, and the deduced mathematical model is too complicated, the dynamical model of MMAV is still a nonlinear system after reasonable simplification. Therefore, this control system is hard to design. The nonlinear predictive control method is proposed to design MMCS, in which controller parameters are optimized based on ACGA. This approach can improve the robustness of MMCS and make it work effectively. The simulation results show that MMCS has good dynamic behavior and strong robustness. As for the mechanism of mass moment control, it is very complicated and the related research is only in the beginning. The public references are insufficient; we only give some useful discussion about the control system design method of MMAV in this paper.

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## Acknowledgments

This work is supported by Natural Science Foundation of Heilongjiang Province of China (F201221), Training Program of Harbin Engineering University for National Natural Science Foundation of China and Fundamental Research Funds for the Central Universities of China (HEUCF100417, HEUCF130402).

## References

- [1] T. Petsopoulos and F. J. Regan, "A moving-mass roll control system for a fixed-trim reentry vehicle," in *Proceedings of the*

*32nd Aerospace Sciences Meeting*, January 1994, AIAA paper no. 94-0033.

- [2] R. Byrne, B. R. Sturgis, and R. D. Robinett, "A moving mass trim control system for reentry vehicle guidance," in *Proceedings of the 21st Atmospheric Flight Mechanics Conference*, 1996, AIAA-96-3438-CP.
- [3] R. D. Robinett III, B. R. Sturgis, and S. A. Kerr, "Moving mass trim control for aerospace vehicles," *Journal of Guidance, Control, and Dynamics*, vol. 19, no. 5, pp. 1064–1070, 1996.
- [4] C. A. Woolsey and N. E. Leonard, "Moving mass control for underwater vehicles," in *Proceedings of the American Control Conference*, pp. 2824–2829, May 2002.
- [5] P. K. Menon, S. S. Vaddi, and E. J. Ohlmeyer, "Finite-horizon robust integrated guidance-control of a moving-mass actuated kinetic warhead," in *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, pp. 5279–5291, August 2006, AIAA 2006-6787.
- [6] J.-W. Li, B.-W. Song, and C. Shao, "Tracking control of autonomous underwater vehicles with internal moving mass," *Acta Automatica Sinica*, vol. 34, no. 10, pp. 1319–1323, 2008.
- [7] X.-Y. Zhang, Y.-Z. He, and Z.-C. Wang, "Robust control of mass moment interception missile based on  $H_\infty$  performance characteristics," *Acta Aeronautica et Astronautica Sinica*, vol. 28, no. 3, pp. 634–640, 2007.
- [8] J. Liu, X. Gao, and Y. Ma, "Study on guidance and control technology of mass moment vehicle," in *Proceedings of the IEEE International Conference on Control and Automation (ICCA '09)*, pp. 679–684, December 2009.
- [9] J. D. Dyer, R. J. Hartfield, G. V. Dozier, and J. E. Burkhalter, "Aerospace design optimization using a steady state real-coded genetic algorithm," *Applied Mathematics and Computation*, vol. 218, no. 9, pp. 4710–4730, 2012.
- [10] Q. Wei, F. Jiancheng, and Z. Weina, "A method of optimization for the distorted model of star map based on improved genetic algorithm," *Aerospace Science and Technology*, vol. 15, no. 2, pp. 103–107, 2011.
- [11] E. T. Turgut and M. A. Rosen, "Relationship between fuel consumption and altitude for commercial aircraft during descent: preliminary assessment with a genetic algorithm," *Aerospace Science and Technology*, vol. 17, no. 1, pp. 65–73, 2012.
- [12] J. F. Wang, J. Periaux, and M. Sefrioui, "Parallel evolutionary algorithms for optimization problems in aerospace engineering," *Journal of Computational and Applied Mathematics*, vol. 149, no. 1, pp. 155–169, 2002.
- [13] A. Shahrokhii and A. Jahangirian, "Airfoil shape parameterization for optimum Navier-Stokes design with genetic algorithm," *Aerospace Science and Technology*, vol. 11, no. 6, pp. 443–450, 2007.
- [14] Y.-H. Li and X.-K. Wei, "Linear-in-parameter models based on parsimonious genetic programming algorithm and its application to aero-engine start modeling," *Chinese Journal of Aeronautics*, vol. 19, no. 4, pp. 295–303, 2006.
- [15] Y.-Z. Luo, G.-J. Tang, and H.-Y. Li, "Optimization of multiple-impulse minimum-time rendezvous with impulse constraints using a hybrid genetic algorithm," *Aerospace Science and Technology*, vol. 10, no. 6, pp. 534–540, 2006.
- [16] R. Fantinutto, G. Guglieri, and F. B. Quagliotti, "Flight control system design and optimisation with a genetic algorithm," *Aerospace Science and Technology*, vol. 9, no. 1, pp. 73–80, 2005.
- [17] Y. V. Pehlivanoglu and B. Yagiz, "Aerodynamic design prediction using surrogate-based modeling in genetic algorithm

- architecture,” *Aerospace Science and Technology*, vol. 23, no. 1, pp. 479–491, 2012.
- [18] Y. V. Pehlivanoglu, “A new vibrational genetic algorithm enhanced with a Voronoi diagram for path planning of autonomous UAV,” *Aerospace Science and Technology*, vol. 16, no. 1, pp. 47–55, 2012.
- [19] S. Kumar and C. S. P. Rao, “Application of ant colony, genetic algorithm and data mining-based techniques for scheduling,” *Robotics and Computer-Integrated Manufacturing*, vol. 25, no. 6, pp. 901–908, 2009.
- [20] P. Hoseini and M. G. Shayesteh, “Efficient contrast enhancement of images using hybrid ant colony optimisation, genetic algorithm, and simulated annealing,” *Digital Signal Processing*, vol. 23, no. 3, pp. 879–893, 2013.
- [21] Y.-H. Zhang, L. Feng, and Z. Yang, “Optimization of cloud database route scheduling based on combination of genetic algorithm and ant colony algorithm,” *Procedia Engineering*, vol. 15, pp. 3341–3345, 2011.
- [22] W. Zhang and T. Lu, “The research of genetic ant colony algorithm and its application,” *Procedia Engineering*, vol. 37, pp. 101–106, 2012.
- [23] Z.-J. Lee, S.-F. Su, C.-C. Chuang, and K.-H. Liu, “Genetic algorithm with ant colony optimization (GA-ACO) for multiple sequence alignment,” *Applied Soft Computing Journal*, vol. 8, no. 1, pp. 55–78, 2008.
- [24] A. Ghanbari, M. Kazemi, F. Mehmanpazir, and M. M. Nakhostin, “A Cooperative Ant Colony Optimization-Genetic Algorithm approach for construction of energy demand forecasting knowledge-based expert systems,” *Knowledge-Based Systems*, vol. 39, pp. 194–206, 2013.
- [25] B. Hemmateenejad, M. Shamsipur, V. Zare-Shahabadi, and M. Akhond, “Building optimal regression tree by ant colony system-genetic algorithm: application to modeling of melting points,” *Analytica Chimica Acta*, vol. 704, no. 1-2, pp. 57–62, 2011.
- [26] S. R. Jangam and N. Chakraborti, “A novel method for alignment of two nucleic acid sequences using ant colony optimization and genetic algorithms,” *Applied Soft Computing Journal*, vol. 7, no. 3, pp. 1121–1130, 2007.
- [27] T. Keskindurk, M. B. Yildirim, and M. Barut, “An ant colony optimization algorithm for load balancing in parallel machines with sequence-dependent setup times,” *Computers and Operations Research*, vol. 39, no. 6, pp. 1225–1235, 2012.
- [28] S. Akpınar, G. M. Bayhan, and A. Baykasoglu, “Hybridizing ant colony optimization via genetic algorithm for mixed-model assembly line balancing problem with sequence dependent setup times between tasks,” *Applied Soft Computing*, vol. 13, no. 1, pp. 574–589, 2013.
- [29] G. Xu, T. Li, X. Zhang, and L. Zhang, “Modeling and motion analysis of a missile based on mass moment control,” *Journal of Harbin Engineering University*, vol. 32, no. 12, pp. 1588–1593, 2011.
- [30] H.-B. Duan, *Ant Colony Algorithms: Theory and Applications*, Science Press, Beijing, China, 2005.
- [31] R. J. Mullen, D. Monekosso, S. Barman, and P. Remagnino, “A review of ant algorithms,” *Expert Systems with Applications*, vol. 36, no. 6, pp. 9608–9617, 2009.
- [32] M. H. Afshar, “A parameter free Continuous Ant Colony Optimization Algorithm for the optimal design of storm sewer networks: constrained and unconstrained approach,” *Advances in Engineering Software*, vol. 41, no. 2, pp. 188–195, 2010.
- [33] G.-Z. Tan and W.-B. Li, “Design of ant algorithm-based optimal PID controller and its application to intelligent artificial leg,” *Journal of Central South University of Technology*, vol. 35, no. 1, pp. 91–96, 2004.
- [34] Y.-H. Zhang, L. Feng, and Z. Yang, “Optimization of cloud database route scheduling based on combination of genetic algorithm and ant colony algorithm,” *Procedia Engineering*, vol. 15, pp. 3341–3345, 2011.

## Research Article

# A Multilayer Hidden Markov Models-Based Method for Human-Robot Interaction

**Chongben Tao and Guodong Liu**

*Key Laboratory of Light Industry Process Advanced Control, School of Internet of Things Engineering, Jiangnan University, Wuxi 214122, China*

Correspondence should be addressed to Chongben Tao; [tom1tao@163.com](mailto:tom1tao@163.com)

Received 31 May 2013; Revised 6 August 2013; Accepted 8 August 2013

Academic Editor: Vishal Bhatnaga

Copyright © 2013 C. Tao and G. Liu. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

To achieve Human-Robot Interaction (HRI) by using gestures, a continuous gesture recognition approach based on Multilayer Hidden Markov Models (MHMMs) is proposed, which consists of two parts. One part is gesture spotting and segment module, the other part is continuous gesture recognition module. Firstly, a Kinect sensor is used to capture 3D acceleration and 3D angular velocity data of hand gestures. And then, a Feed-forward Neural Networks (FNNs) and a threshold criterion are used for gesture spotting and segment, respectively. Afterwards, the segmented gesture signals are respectively preprocessed and vector symbolized by a sliding window and a K-means clustering method. Finally, symbolized data are sent into Lower Hidden Markov Models (LHMMs) to identify individual gestures, and then, a Bayesian filter with sequential constraints among gestures in Upper Hidden Markov Models (UHMMs) is used to correct recognition errors created in LHMMs. Five predefined gestures are used to interact with a Kinect mobile robot in experiments. The experimental results show that the proposed method not only has good effectiveness and accuracy, but also has favorable real-time performance.

## 1. Introduction

Human-Robot Interaction (HRI) is one of the hottest research directions in robot field, and activity recognition is one of the most important markers of HRI. As gesture recognition is a branch of activity recognition, methods used in activity recognition are also suitable for gesture recognition. Generally, depending on sensor types, gesture recognition is divided into two control modes: motion sensor-based [1–3] and vision-based [4, 5]. Both of them have their own characteristics, where gesture recognitions based on motion sensors are easy to capture the motion data of a body, but most of them need to be fixed on human's body, which may make users feel uncomfortable. While vision-based gesture recognition also has some disadvantages. For instance, illumination conditions and background scenery of an environment may cause the difficulty of recognition. Additionally, it is difficult to obtain orientation information from a frame of two-dimensional (2D) image.

According to different gesture recognition methods used in HRI, they can be divided into four categories, which include generative method, discriminative method, heuristic

analysis method and combinations of these methods [6]. Generative method uses generative models for randomly generating observed data, typically given some hidden parameters. It specifies a joint probability distribution over observation and label sequences [7]. Generative method attempts to build a complete description of the input or data space, usually with a probabilistic model. Hidden Markov Models (HMMs) is one of the most popular generative methods, which is a probabilistic model with a specific structure. Once the model is established, it can be learned from data and used to analyze the data. Discriminative method analyzes features extracted from sensor data or segments without considering sequential connections in the data. Common discriminative methods include Neural Networks (NNs) [8], Conditional Random Fields (CRFs) [9] and Support Vector Machines (SVM) [10]. Heuristic analysis method is based on the direct characteristic analysis and description of the data from sensors [11]. However, each individual has different characteristics. Thus, this method is difficult to find a consistency way for observation. Since generative method and discriminative method are both machine learning algorithms, their parameters can be training by using different types of individual



FIGURE 1: (a) Xtion Pro Live sensor, (b) Kinect sensor, and (c) Internal structure of Kinect sensor.

data. However, one drawback of generative approach is that enough data must be available to learn the complete probabilistic representations that are required. Moreover, both of these two types of methods have the problem of high-computational complexity. Currently, researchers have tried to integrate the advantages of different methods to address complex activity recognition problems [12].

In recent years, a new type of RGB-Depth (RGB-D) sensor has appeared, which is not only able to extract RGB feature and depth information with position and direction, but is also able to obtain three-dimensional (3D) acceleration and 3D angular velocity information of the body's motion. Asus Xtion Pro Live sensor and Microsoft Kinect sensor are two types of famous RGB-D sensors [13], as shown in Figures 1(a) and 1(b), respectively. The internal structure of a Kinect sensor is shown in Figure 1(c). Since the advent of Kinect sensor in November 2010, researchers have used it to do extensive researches in the field of Human-Robot Interaction (HRI). Giraldo et al. [14] proposed an approach based on machine learning to identify predefined gestures based on depth images obtained from a Kinect sensor. Additionally, they also adopted a kernel-based learning method to discover the relationship among gestures and samples, which can be used to correctly describe gestures. Ren et al. [15] detected the shape of a hand from color images and depth images obtained from a Kinect sensor and then proposed a shape distance metric method called Finger-Earth Mover's Distance for dissimilarity measurement. Finally, they recognize predefined gestures by template matching. Bauer et al. [16] proposed a Human-Robot Interaction system. Firstly, a person makes a predefined gesture, and then a robot try to understand the action and respond accordingly in realtime.

To overcome the shortcomings of large computation and poor real-time performance, a continuous gesture recognition method is proposed for HRI. The rest of the paper is organized as follows: Section 2 describes a Feed-forward

Neural Networks (FNNs) for gesture spotting and the proposed Multilayer Hidden Markov Models (MHMMs) for continuous gesture recognition. And then gives the proposed continuous hand gesture recognition procedure, which contains hand spotting and gesture recognition based on FNNs and MHMMs. The experimental results are shown in Section 3. Finally, Section 4 shows the summary and then the acknowledgments.

## 2. Mhmm-Based Continuous Gesture Recognition

In this paper, the proposed gesture recognition algorithm consists of two parts: the gesture spotting and segment module and the continuous gesture recognition module. An important problem in gesture recognition is how to distinguish gestures and nongestures, which is called *gesture spotting problem*, namely, how to detect a start point and an end point of a gesture [17]. Firstly, a Feed-Forward Neural Networks (FNNs) is used to detect gestures in an action signal, and then a threshold criterion is used to segment gestures from the signal. Finally, MHMMs is used to recognize segmented continuous gestures.

FNNs are firstly used to distinguish whether a particular action is a gesture. When the FNNs determine a certain action is a gesture, then the output is 1, otherwise the output is 0. Features can be well combined together in FNNs for classification of gestures and nongestures. Additionally, through training, the weights and deviations of FNNs can be optimized. Two counters are used to detect a start point and an end point of a gesture in accordance to a preset threshold value. When the end point of a gesture is detected, the spotting and segment module will trigger the recognition module. As MHMMs is a probabilistic model which has a high computational complexity, FNNs-based segment module will be used as a switch to control data flows, thus computation time can be

saved and the real-time performance of the whole algorithm can be approved. Sequential constraints among gestures will be modeled by MHMMs in recognition module. Forward propagation program in the Bayesian filtering will be used to produce a posterior gesture decision and update the recognition result. In this paper, the flow chart of MHMMs-based continuous gesture recognition method is shown in Figure 2.

**2.1. Gesture Spotting and Segment.** In this paper, a Feed-forward Neural Networks (FNNs) is designed to discover gesture data from signals [18]. The structure of a three-layer FNNs module is shown in Figure 3. The number of input nodes is  $n$ , the input variable of the input layer is:  $Y = (y_1, y_2, \dots, y_n)^T$ , which indicates an original input signal of actions; the output vector of hidden layer is  $a_i^2 = (a_1^2, a_2^2, \dots, a_m^2)^T$ ; the output vector of output layer is  $a_i^3 = (a_1^3, a_2^3, \dots, a_m^3)^T$ , which means a gesture spotting result; the weight matrix from the input layer to the hidden layer is  $W_i^1 = (W_1^1, W_2^1, \dots, W_m^1)$ ; the weight matrix from the hidden layer to the output layer is  $W_i^2 = (W_1^2, W_2^2, \dots, W_m^2)$ ; and Sigmoid function  $f(x) = (1 + e^{-x})^{-1}$  is used as an excitation function.

A 3D acceleration  $\alpha_i = [\alpha_x, \alpha_y, \alpha_z]^T$  is one kind of data extracted by RGB-D sensor, which is trained by FNNs. Let  $a_i^3$  be the output of FNNs:

$$a_i^3 = \text{hard lim} \left( f^2 \left( W^2 f^1 \left( W^1 \mu_i + b^1 \right) + b^2 \right) - 0.5 \right), \quad (1)$$

where  $W^1$ ,  $W^2$ ,  $b^1$ , and  $b^2$  are parameters of FNNs.

Both of the network functions  $f^1$  and  $f^2$  are log-sigmoid functions; function  $\text{hard lim}$  is a hard limiting function. Therefore, parameters of FNNs can be trained by a back-propagation method. In the output layer, set  $W^3 = 1$  and  $b^3 = 0$  in order to produce a discrete output. The output of FNNs is a binary number (1 or 0), which, respectively, represents a gesture or a nongesture. The first layer and the second layer constitute a two-layer feed-forward network which can be obtained optimized parameters through training. And then, weights and deviations are fixed in the output layer in order to produce a discrete output.

Moreover, through training the neural networks, weights and deviations of the FNNs can be optimized. Two digital counters are used for continuous recording of the output of FNNs. When both of them exceed a preset threshold value, the start point and the end point of a gesture can be detected, thereby erroneous classification of the FNNs can be prevented. When an end point of a gesture is inspected, the spotting and segment module will trigger the recognition module.

**2.2. Multilayer Hidden Markov Models.** In our daily life, people usually use some gestures to command pets. And these gestures are corresponded to some reasonable modes which reflect sequential restrictions among continuous gestures. Therefore, these restrictions can be used to improve the accuracy of gesture recognition. According to this inspiration, this paper proposed a continuous gesture recognition algorithm based on MHMMs which is a statistical model

derived from HMMs. HMMs is a famous statistical model for continuous data recognition, which has been widely used in speech recognition, handwriting recognition, and pattern recognition [19]. It is characterized by a set of parameters  $\lambda = (\mathbf{A}, \mathbf{B}, \boldsymbol{\pi})$ , where  $\mathbf{A}$ ,  $\mathbf{B}$  and  $\boldsymbol{\pi}$  are a state transition probability distribution, a probability distribution of observation symbols in each state and an initial state distribution, respectively. In this paper, a forward-backward algorithm is used to estimate a series of likelihood values of observation  $\mathbf{P}(\mathbf{O} | \lambda)$  when given a specific HMMs [20]. For a given observation sequence  $\mathbf{O}$  in the test mode, a Viterbi algorithm is used to find the best state sequence  $\mathbf{Q}$  [21]. An Expectation Maximization (EM) method is used to train the parameters of HMMs [22].

The proposed MHMMs are a kind of generalization for a segmented model, where each segment has subsegments. MHMMs are divided into two layers, where LHMMs are used to identify individual gestures in the continuous gesture signal, and a Bayesian filter with sequential constraints in UHMMs is used to correct recognition errors created in LHMMs. Figure 4 shows the basic idea of MHMMs. A time sequence is layered into several segments, where  $S_i^1$  represents the state sequence of UHMMs and  $S_i^2$  represents the state sequence of LHMMs.  $S_i^2$  is the sub-HMMs of state sequence  $S_i^1$ .  $\mathbf{O}_i^2$  means the decision of LHMMs.

**2.2.1. Data Preprocessing.** Before MHMMs are used to training and recognition gesture data segmented by FNNs, a preprocessing step needs to be completed firstly, which includes removing high-frequency noise and finding the stroke duration of a gesture. Each raw data obtained by a Kinect sensor is composed of a 6-component vector  $\mathbf{U}$ : a 3D acceleration  $[\alpha_x, \alpha_y, \alpha_z]^T$  and a 3D angular velocity  $[\omega_x, \omega_y, \omega_z]^T$ , where

$$\mathbf{U} = [\alpha_x, \alpha_y, \alpha_z, \omega_x, \omega_y, \omega_z]^T. \quad (2)$$

When the robot computer receives a set of data, as the original data contains high-frequency noise, which may affect gesture recognition, a low pass filter with a cutoff frequency of 5 Hz is used to remove the noise and meanwhile smooth the data, and then a sliding window with the length of 100 ms is used to calculate the average to remove the DC components in the 3D acceleration in time domain and generate a vector  $\mathbf{w} = [d_x, d_y, d_z]^T$ . Additionally, a Fast Fourier Transform (FFT) is applied to the deviation vector  $\mathbf{w}$  in frequency domain, and the maximum frequency among  $d_x$ ,  $d_y$ , and  $d_z$  is used as the fundamental frequency of a gesture. Finally, the data point  $\mathbf{V}$  composing of a 3D angular velocity vector, a 3D acceleration vector, and a 3D deviation vector of acceleration is fed into MHMMs for gesture recognition, where

$$\mathbf{V} = [\alpha_x, \alpha_y, \alpha_z, \omega_x, \omega_y, \omega_z, d_x, d_y, d_z]^T. \quad (3)$$

Then a K-means clustering method is used to train centroids in order to quantize vectors into observable symbols. The process of symbolization converts feature vectors into a finite number of symbols, which can be used in the discrete

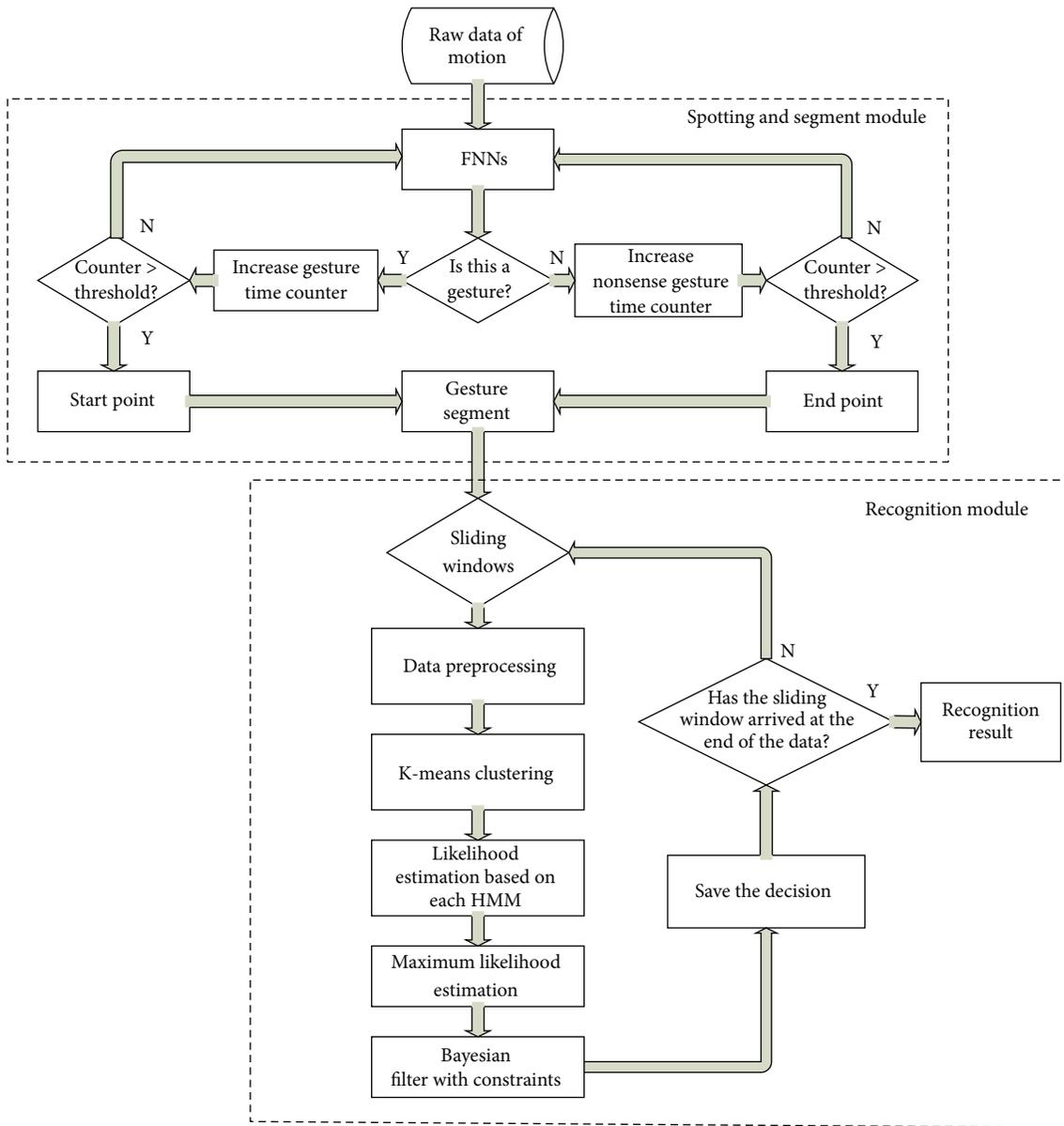


FIGURE 2: Flow chart of MHMMs-based continuous gesture recognition.

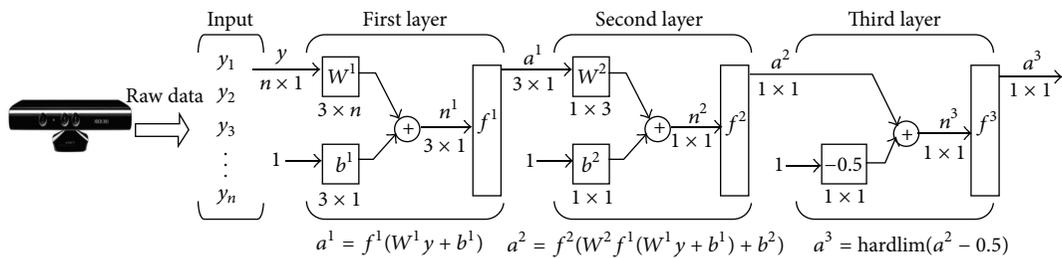


FIGURE 3: The structure of a three-layer FNNs.

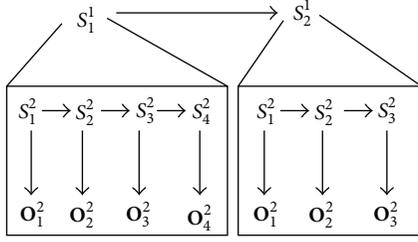


FIGURE 4: The architecture of the proposed MHMMs.

HMMs. Namely, the K-means clustering method is applied on the vector  $\mathbf{V}$  to get the partition value for each vector and also a set of centroids for clustering the data into observable symbols in the recognition phase. The K-means clustering method clusters  $n$  objects into  $k$  partitions, where  $k < n$ , which is based on the properties of objects.

As shown in Figure 5, a sliding window of 1 second (150 data points) is moving with a speed of 30 data points each step, which is simultaneously estimating likelihood of each set of HMMs parameters. Each sliding window generates a likelihood value. Therefore, for a certain gesture, the model which maximizes the likelihood over other HMMs to be the recognized type is chosen as the output decision of the sliding window.

### 2.2.2. LHMMs-Based Individual Gestures Recognition

(1) *Training Phase.* The training phase is divided into four steps, which include finding the stroke duration of a gesture; quantifying the vectors into observable symbols; setting up initial HMMs parameters; and iterating for Expectation Maximization.

*Step 1* (find the stroke duration of a gesture). FFT is used to find the stroke duration of the gesture for HMMs.

*Step 2* (quantify the vectors into observable symbols). After preprocessing the raw data, these data are used for the observation sequence  $\mathbf{O} = \mathbf{O}_1\mathbf{O}_2 \dots \mathbf{O}_T$  and the parameters of HMMs  $\lambda = (\mathbf{A}, \mathbf{B}, \boldsymbol{\pi})$ , respectively. Given parameters of the model, conditional probability of the observation sequence can be calculated. Namely, the problem is given a specific HMMs to assess the probability of an observation sequence  $\mathbf{P}(\mathbf{O} | \lambda)$ , a forward-backward algorithm is used to realize the task as follows [20]. Define the forward variables

$$\alpha_t(i) = \mathbf{P}(\mathbf{O}_1\mathbf{O}_2 \dots \mathbf{O}_t, q_t = S_i | \lambda); \quad (4)$$

that is, given a model, the output observable sequence is  $\mathbf{O}_1\mathbf{O}_2 \dots \mathbf{O}_T$  to the time  $t$ , and the probability of the state is  $S_i$  at time  $t$ . The forward variables can be recursive calculated by the following formula:

$$\alpha_{t+1}(j) = \left[ \sum_{i=1}^N \alpha_t(i) a_{ij} \right] b_j(\mathbf{O}_{t+1}), \quad (5)$$

where  $1 \leq t \leq T-1, 1 \leq j \leq N$ .

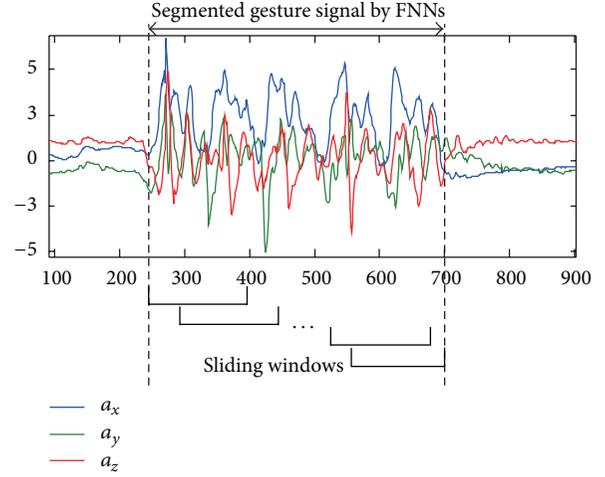


FIGURE 5: Sliding windows for preprocessing.

Define the backward variables

$$\beta_t(i) = \mathbf{P}(\mathbf{O}_{t+1}\mathbf{O}_{t+2} \dots \mathbf{O}_T | q_t = S_i, \lambda). \quad (6)$$

That is, given model parameters at time  $t$ , the state is  $S_i$ . From the moment  $t+1$  to the end of the sequence, the probability of the output observation sequence is  $\mathbf{O}_1\mathbf{O}_2 \dots \mathbf{O}_T$ . Similarly, a recursive method can be used to calculate  $\beta_t(i)$  as follows:

$$\beta_t(i) = \sum_{j=1}^N a_{ij} b_j(\mathbf{O}_{t+1}) \beta_{t+1}(j), \quad (7)$$

where  $t = T-1, T-2, \dots, 1, 1 \leq j \leq N$ .

Then, a K-means clustering method is applied to a 9D vector  $\mathbf{V}$  to get the partition value for each vector and also a set of centroids for clustering the data into observable symbols in the recognition phase.

*Step 3* (set up initial HMMs parameter). Set up state numbers in the model. The initial value of different observable symbols  $\mathbf{O} = \mathbf{O}_1\mathbf{O}_2 \dots \mathbf{O}_T$  and  $\lambda = (\mathbf{A}, \mathbf{B}, \boldsymbol{\pi})$  in each state need to be iterative, which should meet the randomness constraints of HMMs parameters. Then a Viterbi algorithm is used to find the single best state sequence  $\mathbf{Q} = q_1q_2 \dots q_T$  [21].

*Step 4* (expectation maximization iteration). A Baum-Welch algorithm is used to calculate an auxiliary function and maximize the likelihood value of  $\bar{\lambda}$  [22].  $N$  iterations are preceded until the likelihood approaches to a steady value. Expectation value  $\mathbf{Q}(\lambda, \bar{\lambda})$  and the maximized likelihood  $\bar{\lambda}$  are calculated as follows:

$$\mathbf{Q}(\lambda, \bar{\lambda}) = \sum_{\mathbf{Q}} \mathbf{P}(\mathbf{Q} | \mathbf{O}, \lambda) \log [\mathbf{P}(\mathbf{O}, \mathbf{Q} | \bar{\lambda})], \quad (8)$$

$$\max_{\bar{\lambda}} [\mathbf{Q}(\lambda, \bar{\lambda})] \Rightarrow \mathbf{P}(\mathbf{O} | \bar{\lambda}) > \mathbf{P}(\mathbf{O} | \lambda).$$

(2) *Recognition Phase.* During the recognition phase, after training a group of centroids for K-means clustering, several

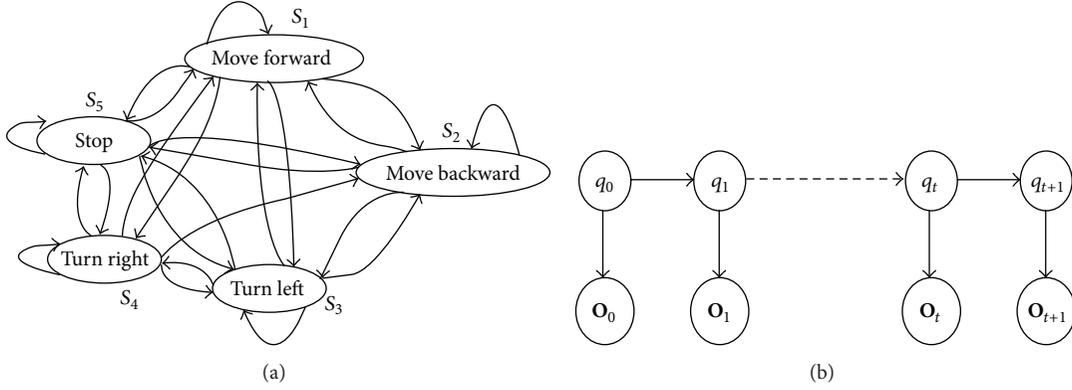


FIGURE 6: (a) transitions among UHMMs consider sequential constraints of context information; (b) update gesture recognition result by UHMMs.

groups of HMMs are formed. Then we calculate the likelihood of each group of HMMs parameters. Given observable sequence  $\mathbf{O}$ , the Viterbi algorithm is used to find the single best state sequence  $\mathbf{Q}$ , and the HMMs with the greatest likelihood can be identified as the most likely gesture type.

**2.2.3. UHMMs-Based Continuous Gesture Update.** In order to improve accuracy in decision making, we consider the sequential conditions of “Context” in UHMMs, where a Bayesian filter is used to update results from LHMMs. Therefore, five gestures are predefined, which includes “Move forward”, “Move backward”, “Turn left”, “Turn right,” and “Stop”. Meanwhile, the “Context” is used as sequential constraints among different types of gestures. For example, the event that someone continuously sends a same command to a robot has a small probability; someone firstly sends a “Move forward” command, and before he wants to send a “Move backward” command, he should send a “Stop” command. Figure 6(a) shows a transition among UHMMs, which is a discrete first-order HMMs with five states and observation symbols, respectively. UHMMs can be described as a sequence of gestures and at any time as being in one of a set of  $N$  ( $N = 5$ ) distinct states:  $S_1, S_2, \dots, S_5$ . The system undergoes a change of state according to a set of probabilities associated with the state. Each arch represents a probability of transition between two adjacent states. The moment associated with change of states is indicated as  $k = 1, 2, \dots, N$  and the  $k$ th actual state is  $q_k$ . The probability  $a_{ij}$  described below relates the current state with the previous state:

$$a_{ij} = \mathbf{P} [q_k = S_j | q_{k-1} = S_i]. \quad (9)$$

The initial state distribution represents the probability distribution of the first order, which is defined as

$$\pi = \mathbf{P} [q_1 = S_i, (i = 1, 2, \dots, N)]. \quad (10)$$

Another element of LHMMs is the state probability distribution of observation symbol  $S_j$

$$b_j(k) = \mathbf{P} [\mathbf{O}_k | q_t = S_j], \quad (11)$$

$b_j$  means the probability which is identified as different observation symbols, where  $\mathbf{O}_k$  represents decisions made by UHMMs.

Figure 6(b) shows a sequence in UHMMs, where the state  $q_t$  represents a  $t$ th gesture, and  $\mathbf{O}_t$  is a majority decision result given by LHMMs. A Forward propagation method [23] is used to update the joint probability observation gathered until time  $t$ . In this paper, the model is defined as

$\mathbf{P}(\mathbf{O}_{t+1} | q_{t+1} = S_j)$  represents  $b_j(\mathbf{O}_{t+1})$ , where the state at time  $t + 1$  is  $S_j$  and the observation is  $\mathbf{O}_{t+1}$ ;

$\mathbf{P}(q_{t+1} = S_j | q_t = S_i)$  means the transition probability  $a_{ij}$  in matrix  $\mathbf{A}$ .

The forward variable  $\alpha_t(i)$  is defined as the probability of the observation sequence  $\mathbf{O}_1 \mathbf{O}_2 \dots \mathbf{O}_t$ , and state  $S_i$  at time  $t$ , given the model  $\lambda$ .

$$\alpha_t(i) = \mathbf{P} (\mathbf{O}_1 \mathbf{O}_2 \dots \mathbf{O}_t, q_t = S_i | \lambda), \quad (12)$$

where  $1 \leq i \leq N$ .

According to the network structure shown in Figure 6(b), we can conclude

$$\begin{aligned} \alpha_{t+1}(j) &= \mathbf{P} (\mathbf{O}_{t+1} | q_{t+1} = S_j) \\ &\times \sum_{S_i} \mathbf{P} (q_{t+1} = S_j | q_t = S_i) \alpha_t(i) \end{aligned} \quad (13)$$

with the initial condition of uniform distribution

$$\alpha_0(i) = \mathbf{P} (q_0 = S_i) = \pi_i. \quad (14)$$

For the UHMMs, the training parameters  $\lambda(\mathbf{A}, \mathbf{B}, \pi)$  are obtained by observing interaction between experimenter and robot, so these parameters vary from person to person. The transition matrix  $\mathbf{A}$  is estimated from the statistical results of actually observed gestures. Probability distribution of observation symbol  $\mathbf{B}$  is an accuracy matrix of individual gestures from LHMMs. The forward variable is updated after having obtained the current observation value, which represents the posterior probability of a current state in the case of given context constraints of UHMMs. Therefore, the updated result is the maximum posteriori probability of the state.

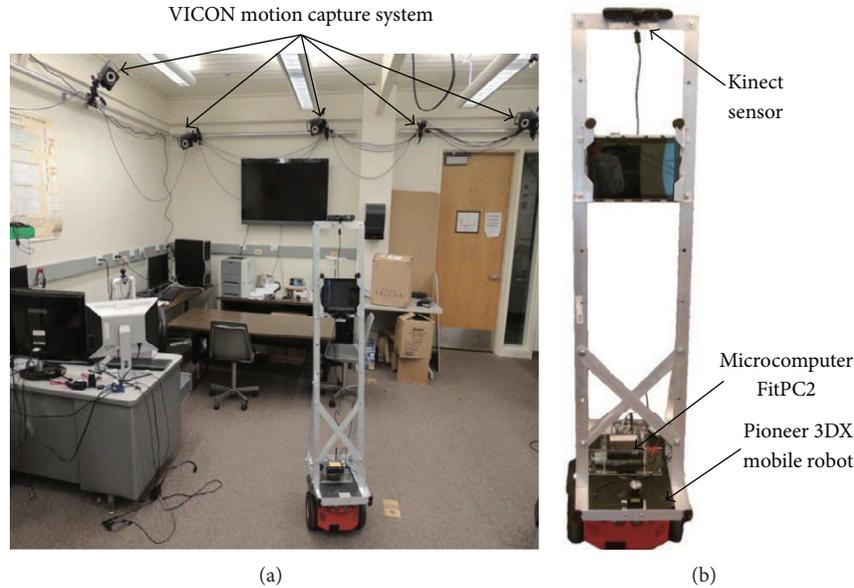


FIGURE 7: Experimental environment and a Kinect mobile robot.

### 3. Experiments

**3.1. Introduction of Hardware Platform.** An experimental environment where a VICON motion capture system [24] and a Kinect mobile robot platform are installed is shown in Figures 7(a) and 7(b), respectively. The mobile robot platform mainly includes: a Pioneer 3DX mobile robot, a microcomputer called FitPC2, and a Kinect sensor which is developed from a RGB camera and a depth sensor. The effective sensing range of the Kinect sensor is from 0.4 meters to 4 meters, vertical viewing angle range is  $\pm 430$ , horizontal range is  $\pm 570$ , and frame rate is 30 fps. Microcomputer FitPC2 is a fanless minicomputer which is able to install Windows or Linux operating systems, and equipped with WiFi. Besides the main components mentioned above, an external battery is used to supply power for both of FitPC2 and Kinect sensor.

Robot Operating System (ROS) is installed in the Linux operating system of FitPC2 microcomputer, which is an open source metaoperating system and provides services like a real operating system, including hardware abstraction, low-level device control, implementation of commonly-used functionality, message-passing between processes, and package management [25]. In this paper, all processes are developed in accordance to the ROS framework, which includes a driver of Kinect sensor, a motion data receiver program, a motion data processing program, and a display program.

**3.2. Motion Data Evaluation Based on VICON.** In order to assess accuracy of the motion data captured by a Kinect sensor, a VICON motion capture system was used to evaluate. As angular velocity data is similar to acceleration data, we select angular velocity for evaluation. Firstly, several markers of the VICON system were fixed on the wrist of an experimenter, and then the experimenter made a variety of predefined gestures in front of the Kinect sensor. Finally, we sent pitch, roll, and yaw data, respectively recorded from

the VICON system and the Kinect sensor into a PC, which was used to plot these data on the same graph, as shown in Figure 8. The data from VICON system was plotted by solid lines, while the data from the Kinect sensor was plotted by dotted lines. From these Figures, we find that the Kinect sensor has accurately measured motions in most of the time, except that a period of time did not match as shown in Figure 8(a), which was due to the arm swings out of the scope of the Kinect sensor. And the mismatch range between these two types of data was always less than 10 degrees. By comparison with the VICON system, we have confirmed that motion data captured from the Kinect sensor meets the accuracy requirement of gesture recognition.

**3.3. Continuous Gesture Recognition Test.** In order to test the performance of the proposed algorithm, we predefine five gestures for experiments. They are “Move forward”, “Move backward”, “Stop”, “Turn left,” and “Turn right”, where the gesture “Waving front and back” means “Move forward”; the gesture “Waving left and right” means “Move backward”; the gesture “Waving up and down” means “Stop”; the gesture “Swing counterclockwise” means “Turn left”; and the gesture “Swing clockwise” means “Turn right”, as shown in Figure 9.

We respectively recorded 10 sets of data from three experimenters for training and testing, and these experiments were carried out in accordance to the following three steps.

*Step 1.* In the scope of a Kinect sensor, repeat gesture Type 1 for 15 times and take a 5 seconds break. Continue performing the rest of the types following in the same way until Type 5 is done, and then save the data.

*Step 2.* In the scope of a Kinect sensor, perform a sequence of 20 gestures with a break of about 2 seconds between gestures, and save the data.

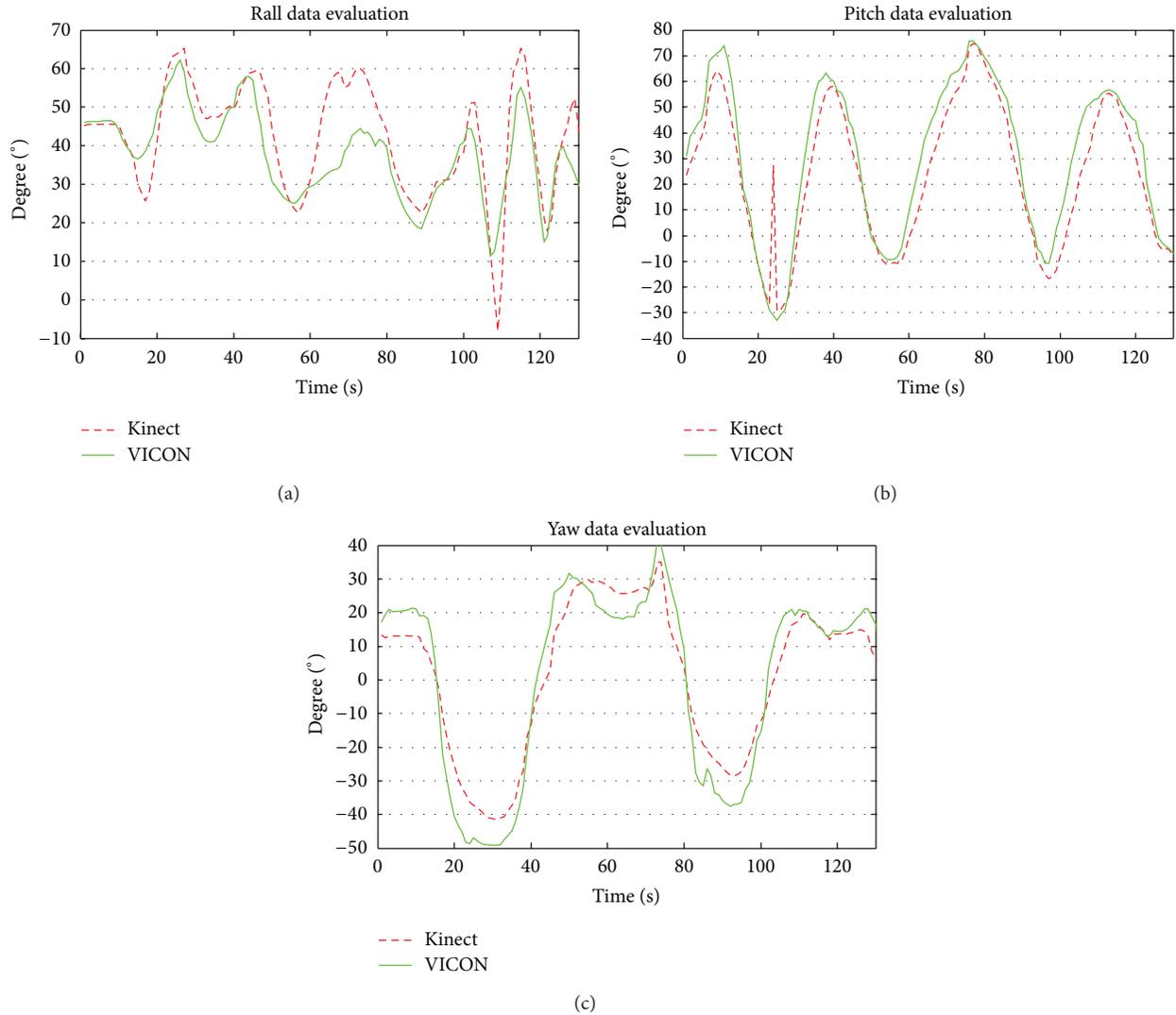


FIGURE 8: Comparisons of Roll (a), Pitch (b), and Yaw (c) data between a VICON system and a Kinect sensor.

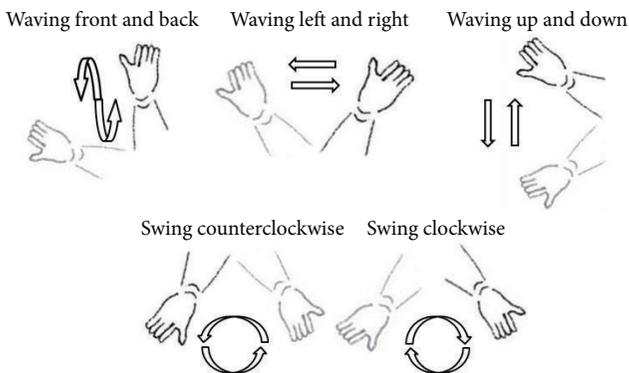


FIGURE 9: Five predefined gestures for Human-Robot Interaction.

*Step 3.* Process the training data and the testing data. Firstly, training the FNNs to distinguish gestures and nongestures; secondly, use each block of training data to train the LHMMs.

To trade off the computational complexity with efficiency and accuracy, set up the number of states 20 in the LHMM. Meanwhile set up the number of distinct observation symbols 20 and then use the trained LHMMs to recognize individual gestures in the test data. The output of each test is a sequence of recognized gestures. Finally, a Bayesian filtering with sequential constraints in UHMMs is used to generate a most likely underlying gesture sequence.

*3.3.1. Performance Evaluation of FNNs.* A MATLAB neural network toolbox [26] is used for training in the first and second layer of FNNs. The initial values of weights and deviations are randomly selected. Different initial values have different performances. If the performance does not reach the target, then the training phase needs to restart a new neural network until it reaches a sufficient accuracy. For instance, within 180 iterations, two different initial values got two different performances, as shown in Figure 10, which gave two results of both good and bad neural network training.

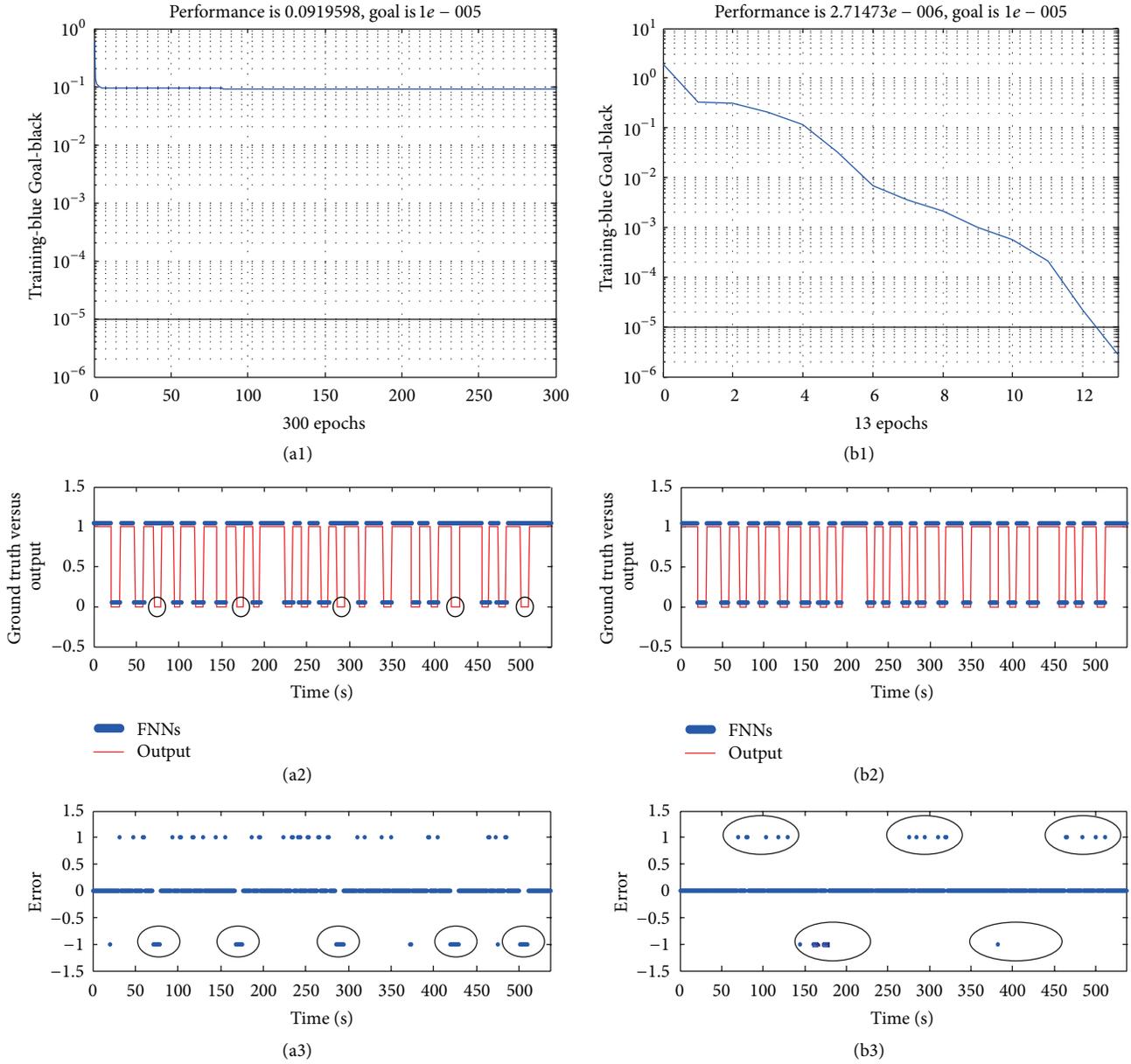


FIGURE 10: Performance comparison of gesture spotting between two FNNs. (a1) the performance goal is not met within 180 iterations. (a2) the output of FNNs, accuracy = 75.69%. (a3) the error of FNNs. (b1) the performance goal is met within 26 iterations. (b2) the output of FNNs, accuracy = 96.86%. (b3) the error of FNNs.

After 180 iterations, the FNNs did not reach the target as shown in Figure 10(a1). These parts circled in Figure 10(a3) showed continuous error, which led to detection errors in Figure 10(a2). Thus, it can be seen that only if the performance curve reaches the standard, the FNNs can achieve sufficient accuracy. As the performance curve cannot meet its target in Figure 10(a1), the training program needs to be restarted. In Figure 10(b1) we find the FNNs achieved the requirement after 26 iterations. Although there were some discrete errors in Figure 10(b3), they did not generate any detection error in Figure 10(b2). Therefore, we can conclude that several discrete errors on the edge of gesture signal do not affect detection results, but consecutive errors can cause detection errors.

**3.3.2. Performance Evaluation of MHMMs.** We respectively recorded 10 sets of data from three experimenters for training and testing again, and every set of sequence data has 20 gestures. The test includes two steps. We firstly performed training LHMMs to recognize individual gestures in a sequence. Secondly, the decision obtained from training was used in a Bayesian filter with sequential restrictions to generate a most likely potential command sequence as the final result. The test results of 3D angular velocity and 3D acceleration are shown in Figures 11 and 12, respectively. In Figure 11(a), the signal was a 3D angular velocity which included twenty gestures captured by a Kinect sensor. While the circled region (A) and (B) in Figure 11(b) indicated two errors generated by FNNs, which caused the size of the segmentation to be shorter

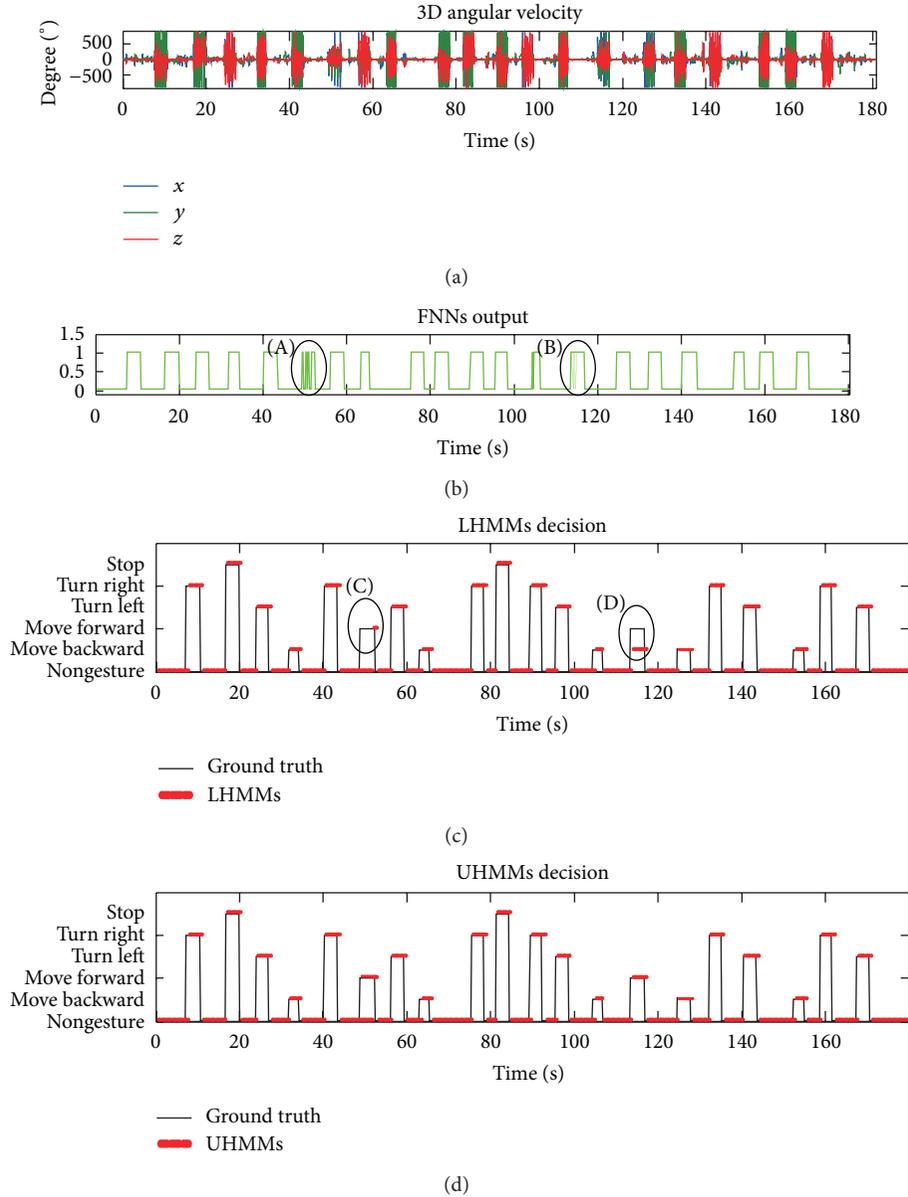


FIGURE 11: Results of FNNs, LHMMs, and UHMMs. (a) the raw 3D angular velocity; (b) the output of FNNs; (c) the LHMMs decision results compared with the ground truth; (d) the UHMMs decision results compared with the ground truth.

than its actual length after detecting the start and end points of the gesture. After using LHMMs to recognize gestures, errors appeared as shown in the circled regions (C) and (D) of Figure 11(c), which indicate the length of segment affects the accuracy of gesture recognition. While, after, UHMMs were used to update recognition results from LHMMs, the output obtained accurate recognition results. Similarly, 3D acceleration data was also used for testing, recognition errors also appeared in LHMMs, as four circled regions shown in Figure 12(c). However, after updating in UHMMs, accurate recognition results were also obtained.

We have calculated the average recognition rate and average time of 30 sets of gestures which were composed by the five predefined gestures. Table 1 lists the average recognition

accuracy and the average recognition time of HMMs and MHMMs, respectively. From the table, we can conclude that the recognition accuracy of MHMMs which use Bayesian filtering is better than that which only uses HMMs, and the real-time performance of the MHMMs-based gesture recognition meets the requirement of online gesture recognition.

#### 4. Conclusion

In this paper, a novel MHMMs continuous gesture recognition algorithm is proposed for Human-Robot Interaction, which uses raw motion data captured by a Kinect sensor. Therefore it is different from traditional vision-based methods. Firstly, a Kinect sensor was used to obtain 3D

TABLE 1: Comparisons of average recognition accuracy and average recognition time between HMMs and MHMMs.

Gesture type	HMMs accuracy (%)	MHMMs accuracy (%)	HMMs recognition time (s)	MHMMs recognition time (s)
1	86.83%	95.71%	0.564	0.637
2	91.24%	97.28%	0.629	0.784
3	78.43%	90.63%	0.468	0.615
4	85.92%	93.42%	0.693	0.812
5	73.97%	89.85%	0.503	0.759

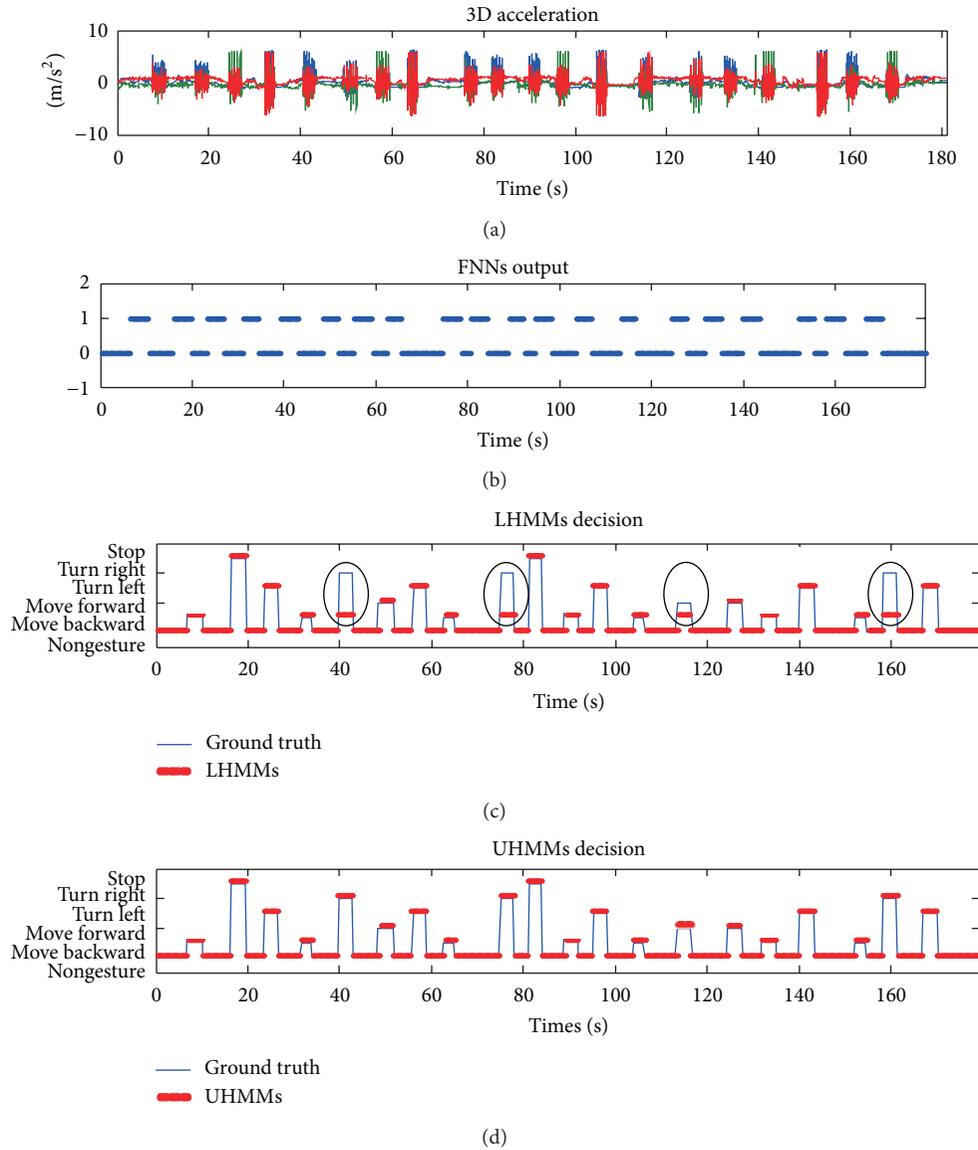


FIGURE 12: Results of FNNs, LHMMs and UHMMs. (a) the raw 3D acceleration; (b) the output of FNNs; (c) the LHMMs decision results compared with the ground truth; (d) the UHMMs decision results compared with the ground truth.

acceleration and 3D angular velocity of an arm. Secondly, a FNNs and a threshold criterion were, respectively, used for gesture spotting and segment. And then the segmented gesture signals were sent into LHMMs to recognize individual gestures. After that, the identified gesture sequence was

fed into UHMMs, where a Bayesian filter with sequential constraints was used to correct errors generated in LHMMs. The experimental results verify the proposed algorithm not only effectively improves the accuracy of continuous gestures, but also has a good real-time performance.

## Acknowledgment

This work is supported by the National Natural Science Foundation of China (60277605).

## References

- [1] X. Wang, M. Xia, and H. Cai, "Hidden markov models based dynamic hand gesture recognition," *Mathematical Problems in Engineering*, vol. 2012, Article ID 986134, 11 pages, 2012.
- [2] A. Pantelopoulos and N. G. Bourbakis, "A survey on wearable sensor-based systems for health monitoring and prognosis," *IEEE Transactions on Systems, Man and Cybernetics C*, vol. 40, no. 1, pp. 1–12, 2010.
- [3] J. Yang, J. Lee, and J. Choi, "Activity recognition based on RFID object usage for smart mobile devices," *Journal of Computer Science and Technology*, vol. 26, no. 2, pp. 239–246, 2011.
- [4] R. Poppe, "A survey on vision-based human action recognition," *Image and Vision Computing*, vol. 42, no. 6, pp. 790–808, 2012.
- [5] D. Weinland, R. Ronfard, and E. Boyer, "A survey of vision-based methods for action representation, segmentation and recognition," *Computer Vision and Image Understanding*, vol. 115, no. 2, pp. 224–241, 2011.
- [6] L. Chen, J. Hoey, and C. D. Nugent, "Sensor-based activity recognition," *Transactions on Systems, Man, and Cybernetics*, vol. 42, pp. 790–808, 2012.
- [7] S. Bilal, R. Akmeliawati, A. A. Shafie, and M. J. E. Salami, "Hidden Markov model for human to computer interaction: a study on human hand gesture recognition," *Artificial Intelligence Review*, vol. 8, pp. 1–22, 2011.
- [8] L. Bao and S. S. Intille, "Activity recognition from user-annotated acceleration data," *Pervasive Computing*, vol. 3001, pp. 1–17, 2004.
- [9] C. Sutton, A. McCallum, and K. Rohanimanesh, "Dynamic conditional random fields: factorized probabilistic models for labeling and segmenting sequence data," *Journal of Machine Learning Research*, vol. 8, no. 2, pp. 693–723, 2007.
- [10] O. Brdiczka, J. L. Crowley, and P. Reignier, "Learning situation models in a smart home," *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 39, no. 1, pp. 56–63, 2009.
- [11] F. Omar, M. Sinn, J. Truszkowski, P. Poupart, J. Tung, and A. Caine, "Comparative analysis of probabilistic models for activity recognition with an instrumented walker," in *Proceedings of the 26th Conference on Uncertainty in Artificial Intelligence (UAI '10)*, pp. 392–400, July 2010.
- [12] F. J. Ordóñez, P. D. Toledo, and A. Sanchis, "Activity recognition using hybrid generative/discriminative models on home environments using binary sensors," *Sensors*, vol. 13, no. 1, pp. 5460–5477, 2013.
- [13] Z. Zhang, "Microsoft kinect sensor and its effect," *IEEE Multi-Media*, vol. 19, no. 2, pp. 4–10, 2012.
- [14] R. Giraldo, D. M. Giraldo, and S. A. Meza, "Kernel based hand gesture recognition using kinect sensor," in *Proceedings of the 17th Symposium of Image, Signal Processing, and Artificial Vision*, pp. 158–161, 2012.
- [15] Z. Ren, J. Meng, J. Yuan, and Z. Zhang, "Robust hand gesture recognition with kinect sensor," in *Proceedings of the 19th ACM International Conference on Multimedia ACM Multimedia 2011 (MM '11)*, pp. 759–760, December 2011.
- [16] A. Bauer, K. Klasing, and G. Lidoris, "The autonomous city explorer: towards natural human-robot interactivity in urban environments," *International Journal of Social Robotics*, vol. 1, no. 3, pp. 127–140, 2009.
- [17] H.-D. Yang, A.-Y. Park, and S.-W. Lee, "Gesture spotting and recognition for human-robot interaction," *IEEE Transactions on Robotics*, vol. 23, no. 2, pp. 256–270, 2007.
- [18] M. T. Hagan, H. B. Demuth, and M. H. Beale, *Neural Network Design*, PWS Publishing Company, 1996.
- [19] L. R. Rabiner, "Tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.
- [20] S.-Z. Yu and H. Kobayashi, "Practical implementation of an efficient forward-backward algorithm for an explicit-duration hidden Markov model," *IEEE Transactions on Signal Processing*, vol. 54, no. 5, pp. 1947–1951, 2006.
- [21] W. Liu and W. Han, "Improved Viterbi algorithm in continuous speech recognition," in *International Symposium in Information Technology*, pp. 542–545, October 2010.
- [22] O. Mourad, "Hmms parameters estimation using hybrid baum-welch genetic algorithm," in *Proceedings of the International Conference on Computer Application and System Modeling*, pp. 207–209, 2010.
- [23] K. Hirasawa, M. Ohbayashi, M. Koga, and M. Harada, "Forward propagation universal learning network," in *Proceedings of the IEEE International Conference on Neural Networks*, pp. 353–358, June 1996.
- [24] VICON, <http://www.vicon.com/>.
- [25] M. Quigley, K. Conley, and P. Gerkey B, "Ros: an open-source robot operating system," in *Proceedings of the ICRA Workshop on Open Source Software*, 2009.
- [26] Neural, <http://www.mathworks.com/products/neuralnet/>.

## Research Article

# Matching Cost Filtering for Dense Stereo Correspondence

Yimin Lin,<sup>1,2</sup> Naiguang Lu,<sup>1,2</sup> Xiaoping Lou,<sup>2</sup> Fang Zou,<sup>3</sup> Yanbin Yao,<sup>3</sup> and Zhaocai Du<sup>3</sup>

<sup>1</sup> Institute of Optical Communication & Optoelectronics, Beijing University of Posts & Telecommunications, Beijing 100876, China

<sup>2</sup> School of Instrumentation Science & Optoelectronics Engineering, Beijing Information Science & Technology University, Beijing 100192, China

<sup>3</sup> Beijing Aeronautical Manufacturing Technology Research Institute, Beijing 100024, China

Correspondence should be addressed to Yimin Lin; linyimin2012@hotmail.com and Naiguang Lu; nglv2002@sina.com

Received 4 July 2013; Accepted 27 August 2013

Academic Editor: Vishal Bhatnaga

Copyright © 2013 Yimin Lin et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Dense stereo correspondence enabling reconstruction of depth information in a scene is of great importance in the field of computer vision. Recently, some local solutions based on matching cost filtering with an edge-preserving filter have been proved to be capable of achieving more accuracy than global approaches. Unfortunately, the computational complexity of these algorithms is quadratically related to the window size used to aggregate the matching costs. The recent trend has been to pursue higher accuracy with greater efficiency in execution. Therefore, this paper proposes a new cost-aggregation module to compute the matching responses for all the image pixels at a set of sampling points generated by a hierarchical clustering algorithm. The complexity of this implementation is linear both in the number of image pixels and the number of clusters. Experimental results demonstrate that the proposed algorithm outperforms state-of-the-art local methods in terms of both accuracy and speed. Moreover, performance tests indicate that parameters such as the height of the hierarchical binary tree and the spatial and range standard deviations have a significant influence on time consumption and the accuracy of disparity maps.

## 1. Introduction

Stereo correspondence between stereo images results in a depth image, also called a disparity map, which can be categorized as sparse or dense. Sparse disparity maps are obtained mainly using feature-based methods derived from human vision research [1]. As a result, high processing speeds and accurate disparity maps are achieved but without high density, which has limited their use for many purposes. Dense stereo correspondence, which aims to figure out which parts of an image correspond to which parts of another image, is a challenging issue in the field of computer vision. The requirement of dense disparity maps is motivated by many contemporary applications such as virtual reality, view synthesis, and robot vision navigation [2].

Dense stereo correspondence algorithms can be classified as global or local according to whether they obtain disparities from global or local information. The goal of global methods (energy based) is to minimize a global cost function which

combines matching costs and smoothness terms, depending on information derived from the whole image. These methods are time consuming but very accurate [3]. On the other hand, local methods (area based) offer high speed at the expense of matching accuracy and determine the degree of disparity of each pixel according to information provided by its local and neighboring pixels. These methods are also referred to as window-based methods because the disparity computation between two matching pairs depends only on the intensity values within a fixed-size and fixed-shape matching window [4]. However, recent studies have shown that, by ingeniously selecting and aggregating the matching costs of neighboring pixels, the disparity maps produced by a local approach can be more accurate than those generated by global methods [5]. The most noteworthy technique is local filtering, which is an effective way to reduce matching noise and is able to generate high-quality disparity maps.

This paper proposes a dense stereo correspondence approach very similar to the original adaptive support weight

(ASW) method [6] to obtain accurate disparity maps both in depth discontinuities and smooth regions. The basic idea is to accept similar pixels within a matching window by assigning them relatively large support weights and to reject dissimilar pixels by giving them very small support weights. Therefore, it is necessary to divide the neighboring pixels into similar and dissimilar groups. In the present case, adaptive support weights are computed from the color image using a hierarchical clustering algorithm inspired by Gastal's work [7] in high-dimensional filtering of images and videos in real time; the disparity maps after filtering are less noisy, and the depth discontinuity boundaries are preserved fairly well. In addition, the proposed algorithm has improved the results for efficiency and accuracy compared with the guided-image filter (GIF) [8] algorithm used for stereo correspondence, which is by far the best existing algorithm.

The main contributions of this paper include the following.

- (1) A novel matching-cost filtering model is proposed based on an edge-preserving filter for which the adaptive support weights are computed using a hierarchical clustering algorithm (as shown in Section 3.2). This solution can reduce mismatching, especially around regions of depth discontinuities, and can reconstruct dense high-accuracy disparity maps.
- (2) The computational complexity of the proposed method is essentially linear both in the number of image pixels and the number of clusters, regardless of the matching window size and the intensity range (as described in Section 3.3). Therefore, the method can be easily adjusted to meet real-time requirements with the help of contemporary graphics hardware (a graphics processing unit (GPU)).
- (3) A new disparity refinement method is presented, which has been proved to be robust and effective for improving the accuracy of coarse disparity maps (as presented in Section 3.5). This method can be applied to other coarse-to-fine frameworks, which are among the classic, simplest, and most popular stereo matching algorithms.
- (4) The influence of algorithm parameters on accuracy and efficiency is discussed, especially regarding the weight coefficient, the height of the hierarchical binary tree, and the size of the spatial and range standard deviations (as discussed in Section 4.2). This study offers recommendations which can be used as a basis for future practical applications.

The rest of this paper is organized as follows: Section 2 describes an overview of the state-of-the-art local filtering methods and our method will be proposed in Section 3. Section 4 presents experimental results which compare the proposed method with other state-of-the-art approaches and discusses the influences of parameter settings. Finally, conclusions and suggestions for future work are discussed in Section 5.

## 2. Related Work

A disparity map is obtained by determining the disparity which has the lowest matching cost in each local matching window, a method which is widely used in local algorithms. Many local methods have been proposed to obtain a dense disparity map recently. For instance, adaptive-window methods [9, 10] try to find an optimal matching window for each pixel, and multiple-window methods [11] select an optimal matching window among predefined multiple windows located at different positions with the same shape. However, these methods have one limitation in common: the shape of the matching window is constrained to be a rectangle, which is not appropriate for pixels near depth discontinuities. Therefore, it is difficult to find an optimal matching window with an appropriate size and shape for all cases.

Instead of searching for an optimal matching window of arbitrary size and shape, it is possible to aggregate costs after local smoothing within a matching window to reduce matching noise. It is clear that most noise can be reduced effectively by a linear filter, such as Gaussian filter, but the disparity map always results in a well-known "edge-fattening" phenomenon. Therefore, the local filtering results will not be a good neighborhood representative close to an edge region. To address this problem, the recently proposed ASW algorithm [6] smoothes the matching costs with an adaptive weighted filter in which the support weights are chosen according to both the color similarity and the Euclidean distance to the center pixel. These methods imitate the way that humans assign different weights to a pixel according to color or brightness in the process of finding the correspondences between their two eyes. Such a filter is also referred to as an edge-preserving filter in computer vision and is widely used for image denoising; examples include the SUSAN filter [12], bilateral filter [13], and the nonlocal means filter [14, 15]. Experimental results show that this approach can produce disparity maps better than those generated using global optimization techniques without needing many user-specified parameters. Although this method leads to high-quality results, its computational speed presents a problem because runtime is computationally expensive. Therefore, many improved and real-time solutions have been presented, such as the  $O(1)$  bilateral filter [16–18], the dual-cross-bilateral grid (DCBG) [19, 20], the GIF [21, 22], and the nonlocal filter [23].

## 3. Cost Aggregation with Local Filtering

A literature review has provided a taxonomy and an evaluation of typical matching algorithms and has emphasized that such a coarse-to-fine algorithm generally performs the following four steps [24]:

- (1) cost initialization, in which the matching costs for assigning different disparity hypotheses to different pixels are calculated;
- (2) cost aggregation, in which the initial matching costs are aggregated spatially over matching windows;

- (3) disparity optimization, in which a cost function is minimized to obtain the best disparity hypothesis for each pixel;
- (4) disparity refinement, in which the coarse disparity maps are postprocessed to remove mismatches or to generate fine disparity maps.

According to these four steps, in this paper, the cost aggregation with local filtering consists of five parts: matching cost initialization, cost aggregation with filtering, clustering range values for the sampling points, disparity selection, and refinement. In addition, the computational complexity is discussed.

**3.1. Cost Initialization.** Generally, it is possible to identify matching pairs in stereo images by measuring their similarity. The most common algorithms which use a matching cost function to establish a correspondence between the two points are the sum of absolute intensity differences (SAD), the sum of squared intensity differences (SSD), and the normalized cross-correlation (NCC) [25].

The cost initialization module computes the initial matching cost  $M(u, v, d)$  for assigning disparity hypothesis  $d$  to image pixel  $(u, v)$ , where  $u, v$  define the displacements in the  $x$ - and  $y$ -directions, respectively. Generally, after rectifying a stereo image, there is no shift in the  $y$ -direction except for the displacement in the  $x$ -direction, in which case the cost can be represented as  $M(u, d)$  according to the disparity  $d$ . The costs are calculated using the truncated absolute differences in range (intensity or color) and the gradient between corresponding pixels. In other words,

$$\begin{aligned} M(u, d) &= a \cdot \min [ |I_L(u) - I_R(u - d)|, \tau_1 ] \\ &+ (1 - a) \cdot \min [ |\nabla_x I_L(u) - \nabla_x I_R(u - d)|, \tau_2 ], \end{aligned} \quad (1)$$

where  $a$  is the weight coefficient,  $I_L(u)$  is the left image, and the corresponding right image which has disparity  $d$  is  $I_R(u - d)$ .  $\nabla_x$  is the gray-scale gradients in the  $x$ -directions, and  $\tau_1, \tau_2$  are truncation values for balancing the range and gradient terms. Such a matching cost model has been proved to be robust to illumination changes and is commonly used in stereo correspondence [26].

**3.2. Cost Aggregation with Filtering.** The original local filtering approach tried to compute the weights which are the average of the adjacent matching costs. The costs aggregated over the weights can therefore be expressed as

$$C(i, d) = \frac{\sum_{j \in N_i} W(i, j) M(j, d)}{\sum_{j \in N_i} W(i, j)}, \quad (2)$$

where  $i$  and  $j$  are pixel indices in the  $x$ -direction and  $N_i$  is the region around the  $i$ th coordinate.

The weights  $W(i, j)$  of this linear combination are given by two Gaussian filter kernels which combine the spatial weights based on the distance between two pixels and

the range weights based on the intensity difference. Therefore, the filter weights  $W(i, j)$  can be represented by spatial and range terms as

$$W(i, j) = \exp\left(-\frac{|i - j|^2}{\sigma_s^2}\right) \exp\left(-\frac{|I_i - I_j|^2}{\sigma_r^2}\right), \quad (3)$$

where  $\sigma_s$  and  $\sigma_r$  are two constants used to adjust the spatial and range similarities.

The Gaussian over the range similarity  $R$  can be rewritten as a convolution using two Gaussian kernels:

$$\begin{aligned} R &= \exp\left(-\frac{|I_i - I_j|^2}{\sigma_r^2}\right) \\ &= C \int \exp\left(-\frac{|I_i - m|^2}{\sigma_r^2/2}\right) \exp\left(-\frac{|m - I_j|^2}{\sigma_r^2/2}\right) dm, \end{aligned} \quad (4)$$

where  $C$  is a normalization factor and  $m$  is a sampling range value. Finally, the range  $R$  for a Gaussian integral can be evaluated numerically using an approximation according to the Gauss-Hermite quadrature rule as

$$R = C \sum_{n=1}^K \exp\left(-\frac{|I_i - m_n|^2}{\sigma_r^2/2}\right) \exp\left(-\frac{|m_n - I_j|^2}{\sigma_r^2/2}\right), \quad (5)$$

where  $K$  is the number of sampling range values. Increasing the number of sampling points gives a better approximation for the integral in (4). Assuming that pixel  $i$  has a sampling set  $\{m_{1i}, m_{2i}, \dots, m_{Ki}\}$ , the filter weights in (3) can be rewritten as

$$\begin{aligned} W(i, j) &= \exp\left(-\frac{|i - j|^2}{\sigma_s^2}\right) \\ &\times \sum_{n=1}^K \exp\left(-\frac{|I_i - m_{ni}|^2}{\sigma_r^2/2}\right) \exp\left(-\frac{|m_{ni} - I_j|^2}{\sigma_r^2/2}\right). \end{aligned} \quad (6)$$

The normalization factor  $C$  was not included because both of the numerator and denominator in (2) contain this factor and it will cancel out after the division.

**3.3. Clustering Range Value for Sampling Points.** As mentioned before, the key point of Yang's algorithm [23] is that it accepts similar pixels within a matching window by assigning them relatively large support weights and rejects dissimilar pixels by giving them very small support weights. Clearly, it is necessary to divide neighboring pixels into similar and dissimilar groups. Inspired by this opinion, the authors propose a hierarchical clustering algorithm similar to that developed by Gastal and Oliveira [7] to separate iteratively the whole set of image pixels from different range values into different clusters and to perform cost aggregation with local filtering within these clusters. This is actually an expansion

of the method of adaptive manifold filtering in stereo correspondence and results in a modified clustering algorithm.

Assume that pixel  $i$  and its neighboring pixel  $j$  within a cluster, where their  $n$ th sampling points have similar range values, satisfy

$$m_{ni} = m_{nj}. \quad (7)$$

Averaging values only from pixels belonging to the same cluster generates better estimates for the local filtering output. Therefore, after clustering range values for the sampling points, the cost aggregation in (2) can be rewritten using the filter weights in (6) and the cluster constraints in (7) as

$$\begin{aligned} C(i, d) = & \left( \sum_{n=1}^K \exp\left(-\frac{|I_i - m_{ni}|^2}{\sigma_r^2/2}\right) \right. \\ & \times \sum_{j \in N_i} \exp\left(-\frac{|i - j|^2}{\sigma_s^2}\right) \\ & \times \exp\left(-\frac{|m_{nj} - I_j|^2}{\sigma_r^2/2}\right) M(j, d) \Big) \\ & \times \left( \sum_{n=1}^K \exp\left(-\frac{|I_i - m_{ni}|^2}{\sigma_r^2/2}\right) \right. \\ & \times \sum_{j \in N_i} \exp\left(-\frac{|i - j|^2}{\sigma_s^2}\right) \\ & \times \exp\left(-\frac{|m_{nj} - I_j|^2}{\sigma_r^2/2}\right) \Big)^{-1}. \end{aligned} \quad (8)$$

Compared with the complexity of the original bilateral filter in (3), the proposed filter in (8) reduces the complexity from  $O(N^2)$  to  $O(KN)$ , where  $K \ll N$  and  $N$  is the number of pixels within the whole image.

After introducing the improved cost aggregation and complexity analysis, an algorithm for clustering the range values can be summarized as follows.

*Step 1.* Generate the first sampling point  $m_{1i}$  at pixel  $i$  by low-pass filtering the input signal  $I$  within neighborhood  $N_i$ :

$$m_{1i} = \frac{\sum_{k \in N_i} \exp(-|i - k|^2/\sigma_s^2) I_{i,k}}{\sum_{k \in N_i} \exp(-|i - k|^2/\sigma_s^2)}, \quad (9)$$

where  $I_{i,k}$  represents the range value with distance  $k$  around pixel  $i$ .

*Step 2.* Generate the  $n$ th ( $1 < n \leq K$ ) sampling point  $m_{ni}$ . The first step is to compute an optimal hyperplane  $R_n$ ,

$$\Sigma \cdot R_n = \lambda_{\max} R_n, \quad \text{where } \lambda_{\max} = \max[\text{eigenvalue}(\Sigma)], \quad (10)$$

which corresponds to the eigenvector associated with the largest eigenvalue of the covariance matrix:

$$\Sigma = \frac{1}{N} \sum_{i=0}^{N-1} (x_i - u)(x_i - u)^T, \quad (11)$$

where  $x_i$  is the difference between the range value  $I_i$  and the previous sampling point  $m_{n-1,i}$  associated with each pixel  $i$ :

$$x_i = I_i - m_{n-1,i}, \quad (12)$$

and  $u$  is equal to the sum of the values  $x_i$  divided by the number of pixels  $N$ :

$$u = \frac{1}{N} \sum_{i=0}^{N-1} x_i. \quad (13)$$

*Step 3.* Segment the pixels into two clusters  $C^+$  and  $C^-$  using the sign of the projection:

$$P_i = R_n^T x_i \in \begin{cases} C^+, & P_i \geq 0 \\ C^-, & P_i < 0. \end{cases} \quad (14)$$

*Step 4.* Compute a new sampling point  $m_{ni}^+$  also by low-pass filtering the input signal, but giving weight zero to pixels not in  $C^+$ , as

$$\begin{aligned} m_{ni}^+ = & \left( \sum_{k \in (N_i \cap C^+)} (1 - w_{nk}) \right. \\ & \times \exp\left(-\frac{|i - k|^2}{\sigma_s^2}\right) I_{i,k} \Big) \\ & \times \left( \sum_{k \in (N_i \cap C^+)} (1 - w_{nk}) \right. \\ & \times \exp\left(-\frac{|i - k|^2}{\sigma_s^2}\right) \Big)^{-1}. \end{aligned} \quad (15)$$

The values  $w_{nk}$  are the weights calculated using the range value and the previous sampling points:

$$w_{nk} = \exp\left(-\frac{|I_{i,k} - m_{n-1,k}|^2}{\sigma_r^2/2}\right). \quad (16)$$

Perform the same processing for  $m_{ni}^-$  using pixels belonging to  $C^-$ ; then the combination of  $m_{ni}^+$  and  $m_{ni}^-$  is the whole set of sampling points  $m_{ni}$ .

*Step 5.* The number of sampling range values  $K$  determines whether more clusters are needed for sampling points. Therefore, the next step is to repeat recursively Step 2 onwards until  $n = K$ .

Remember that Steps 2 and 3 can be directly rewritten using the sign (positive or negative) of the differences when the range value is a gray one:

$$P_i = I_i - m_{n-1,i} \in \begin{cases} C^+, & P_i \geq 0 \\ C^-, & P_i < 0. \end{cases} \quad (17)$$

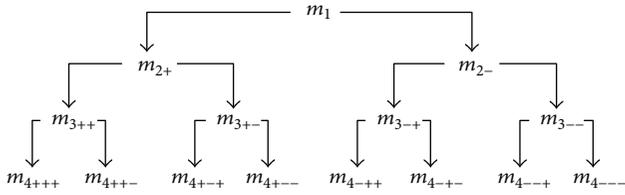


FIGURE 1: Hierarchical binary tree generated by the clustering algorithm.

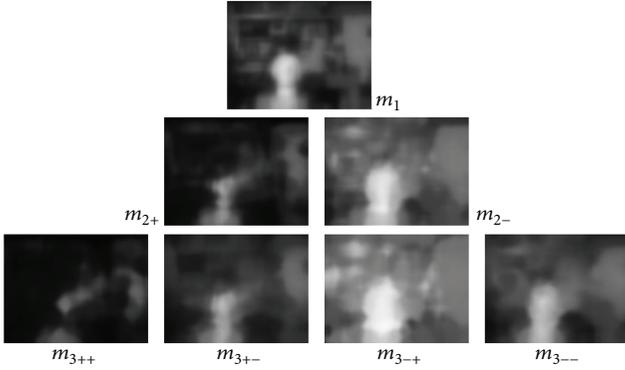


FIGURE 2: The first three levels of sampling points  $m_{ni}$  after clustering the range values.

These five steps serve to construct the hierarchical binary tree shown in Figure 1. The whole tree with height  $H$  has  $K = 2^H - 1$  nodes. Each sampling point  $m_n$  has  $2^n - 1$  nodes. For example, if  $n = 3$ , the third  $m_3$  has four nodes  $\{m_{3++}, m_{3+-}, m_{3-+}, m_{3--}\}$ ; note that the first subscript plus or minus is the same as the upper nodes and the second nodes generated by the current clustering procedure. The rest can be generated in the same manner.

At the top of the tree, the sampling points are better adapted to smooth regions. Points further down this tree would become gradually better adapted to edge regions.

Figure 2 shows the first three levels of sampling points  $m_{ni}$  of the Tsukuba image which was downloaded from the Middlebury benchmark database [27]. Based on clustering range values for more than one sampling points, the filtering results of (8) can be guaranteed to be an edge-preserving smoothing.

**3.4. Disparity Optimization.** Once the matching costs have been filtered using a cluster method, the disparity optimization step computes an optimal disparity map  $D(u, v)$  using the local winner-takes-all (WTA) approach, which computes the coarse disparities associated with the minimum cost value at each pixel. In other words,

$$D(u, v) = \arg \min_{d \in L} C(u, v, d), \quad (18)$$

where  $C(u, v, d)$  represents the matching cost obtained after cost aggregation for assigning a disparity hypothesis to pixel  $(u, v)$  and  $L$  is the number of disparity levels.

**3.5. Disparity Refinement.** The coarse disparity maps generated by WTA may contain some mismatches because local optimization does not obey the smoothness constraint. Therefore, a two-step postprocessing method for fine disparity maps is proposed.

The first step is a left and right cross-checking procedure for mismatches. Two corresponding disparity maps with the left and the right images as reference images are obtained. Then the left and right consistency check divides all the pixels into stable or unstable pixels. Note that all stable pixels in the left and right disparity maps have the same disparity value and that the rest of the pixels are labeled as unstable, represented by a value of zero for all disparity levels.

Secondly, let  $D(u, v)$  represent the left disparity map; a new disparity space volume (DSV) [28] is then computed for each stable (S) or unstable (U) pixel  $(u, v)$  at each disparity level  $d$  as

$$\text{DSV}(u, v, d) = \begin{cases} |d - D(u, v)| & (u, v) \in S \\ 0 & (u, v) \in U. \end{cases} \quad (19)$$

Then an edge-preserving filter such as GIF is applied to smooth the DSV at each disparity level, and the unstable pixels are assigned a new disparity value which depends on the lowest value of the DSV.

## 4. Experimental Results

In this section, the performance of the proposed method is evaluated using the Middlebury stereo benchmark, which provides stereo images with known ground truth [27]. The experimental results are then compared with other local filtering methods which have recently been proven to be the best edge-preserving local stereo methods in terms of both speed and accuracy on the Middlebury benchmark website. Therefore, the comparison results will serve to demonstrate that the proposed method performs well among all local stereo correspondence algorithms. Moreover, this section analyzes the impacts of different parameter settings on the computational complexity and accuracy of the dense disparity maps.

The proposed method was run with constant parameter settings for all four testing images:  $\{a, \tau_1, \tau_2, H, \sigma_r, \sigma_s\} = \{0.1, 0.028, 0.08, 4, 0.08, 11\}$ . To analyze and compare the quality of the stereo matching algorithms, a widely accepted quantitative performance evaluation criterion, the percentage of bad pixels (PBP), was introduced:

$$\text{PBP} = \left[ \frac{1}{N} \sum_{x,y} (|d_t(x, y) - d_g(x, y)| > \delta) \right] \times 100\%, \quad (20)$$

where  $N$  is the total number of pixels,  $d_t$  and  $d_g$  are the computed depth mapping and the ground truth mapping, and  $\delta$  is an absolute disparity error threshold. A value of  $\delta = 1$  was chosen in these experiments because this setting is the same as in some previously published studies. Hence, a smaller PBP number means a better-performing algorithm. The preferred metric (PBP) used in this paper, which is

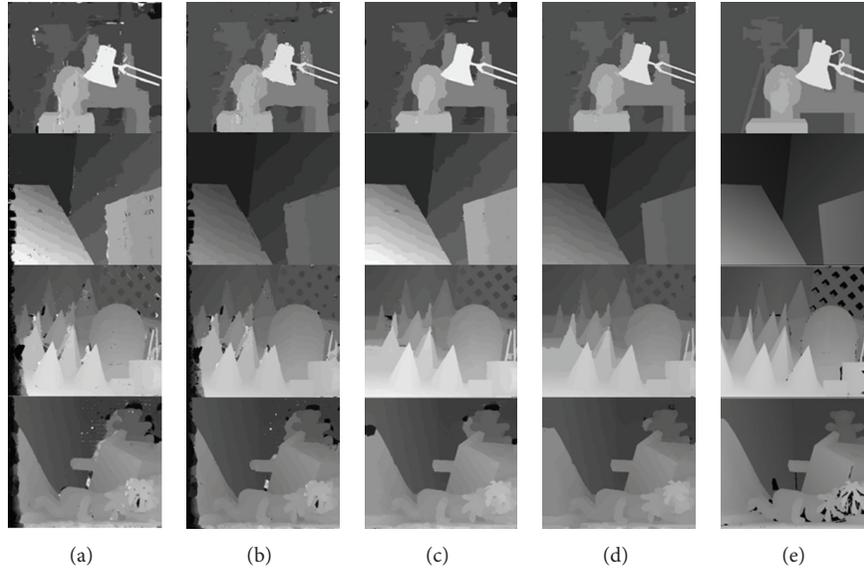


FIGURE 3: Experimental results on the Middlebury benchmark. Dense disparity maps from the first to the last row are the “Tsukuba,” “Venus,” “Cones,” and “Teddy” images. ((a) and (b)) The results of GIF and the proposed method without refinement procedure. ((c) and (d)) The disparity maps obtained using the GIF and the proposed method with refinement procedure. (e) Ground truth.

TABLE 1: Quantitative evaluation for the Middlebury image pairs.

Method	Tsukuba			Venus			Teddy			Cones			AE
	Non	All	Disc	Non	All	Disc	Non	All	Disc	Non	All	Disc	
HCR	1.56	1.78	8.07	0.22	0.34	2.96	6.36	11.93	15.62	2.88	8.14	8.22	<b>5.67</b>
GIR	1.87	2.23	7.92	0.27	0.47	2.60	6.74	12.28	16.20	2.94	8.35	8.36	<b>5.85</b>
ASW	1.38	1.85	6.90	0.71	1.19	6.13	7.88	13.30	18.60	3.97	9.79	8.26	<b>6.66</b>
HCN	2.14	2.94	9.16	1.25	1.94	9.36	7.22	15.28	17.96	3.41	12.93	9.61	<b>7.77</b>
GIN	2.53	3.32	8.63	1.98	3.13	15.81	8.35	16.87	18.81	3.64	12.64	9.70	<b>8.78</b>
DCBG	5.90	7.26	21.0	1.35	1.91	11.20	10.5	17.2	22.2	5.34	11.9	14.9	<b>10.89</b>

considered the most representative of the quality of the results, will be used to make comparison easier.

#### 4.1. Comparison of Disparity Maps

**4.1.1. Accuracy of the Dense Disparity Maps.** The GIF-based cost-aggregation method and the proposed hierarchical clustering method were first used to aggregate matching costs. Then winner-take-all and refinement operations were used to obtain the dense disparity maps. As shown in Figure 3, both methods yielded accurate results for the depth discontinuities as well as in the smooth regions for the test images.

The corresponding quantitative results are presented in Table 1, which records PBP in the nonoccluded, depth-discontinuous, and overall regions of the “Tsukuba,” “Venus,” “Cones,” and “Teddy” images. The rightmost column of the table contains the average errors (AE), which were calculated using the average PBP over all twelve columns. As can be seen from the fourth and fifth rows of Table 1, the AE values obtained using the GIF (GIN) and the proposed method (HCN) without the refinement procedure were 8.78% and 7.77%, respectively. The first two rows show the errors

obtained using the GIF (GIR) and the proposed method (HCR) with the refinement procedure; the AE values were 5.85% and 5.67%, respectively. This shows that the proposed method outperformed the GIF for filtering matching costs during cost aggregation. As expected, the proposed refinement method is suitable for removing mismatches, and the improvement is evident. In particular, as can be seen in Table 1, HCR can also outperform the original ASW algorithm [6] and the fast DCBG technique [20]. In the authors’ opinion, the method proposed in this research may well achieve the topmost position among local stereo correspondence algorithms.

To verify algorithm stability, the performance of the GIF and the proposed methods was compared on an additional 27 Middlebury stereo images [27]. As described above, the PBP values with a disparity error larger than one pixel in all the regions were used to build the average of this measure over all 27 test images. The corresponding quantitative evaluation is summarized in Table 2. Note that both methods may be less accurate in large untextured regions such as the Middl and Monopoly pairs. Errors in untextured regions are due mostly to mismatches and will cause inconsistencies between the left

TABLE 2: Evaluation for stereo methods on all 27 Middlebury stereo pairs.

Method	Aloe	Baby1	Baby2	Baby3	Bowling1	Bowling2	Cloth1	Cloth2	Cloth3	Cloth4
HCN	12.71	11.14	11.81	17.87	26.70	19.10	9.71	16.37	11.15	14.95
GIN	13.42	12.39	12.88	17.98	27.37	19.19	10.36	16.56	11.30	15.40
HCR	8.19	4.99	7.24	9.74	20.69	14.38	4.61	10.96	5.34	10.66
GIR	8.78	5.41	7.59	9.99	20.26	14.61	5.03	11.43	5.43	10.81
Method	Flowerpots	Lampshade1	Lampshade2	Midd1	Midd2	Monopoly	Plastic	Rocks1	Rocks2	Wood1
HCN	23.60	23.03	30.95	45.66	41.66	36.51	43.62	11.90	12.24	16.78
GIN	23.41	24.13	32.64	46.58	42.90	34.78	47.81	11.72	11.83	17.61
HCR	18.48	15.86	23.46	43.95	37.30	22.71	35.60	5.72	5.19	5.26
GIR	18.81	16.67	24.01	44.35	38.62	25.01	38.33	5.55	5.02	5.57
Method	Wood2	Art	Books	Dolls	Laundry	Moebius	Reindeer	AE		
HCN	15.43	26.26	21.59	17.32	27.98	20.32	21.96	<b>21.79</b>		
GIN	14.83	26.41	21.10	16.68	29.19	20.06	23.12	<b>22.28</b>		
HCR	0.64	18.60	17.85	11.90	20.80	14.68	8.19	<b>14.92</b>		
GIR	0.57	18.59	17.50	11.96	22.80	14.22	8.36	<b>15.38</b>		

TABLE 3: Run time comparison of the GIF and the proposed method in seconds.

Version	Method	Tsukuba	Venus	Teddy	Cones
CPU	GIF	32	80	251	257
	HC	28	62	204	203
GPU	GIF	0.330	0.543	1.677	1.695
	HC	0.270	0.434	1.307	1.315

and right disparity maps. However, the proposed HC method is still the winner and slightly outperforms the GIF technique. In a comparison of HCN and HCR, the proposed refinement method is expected to perform well.

**4.1.2. Computational Complexity.** We have implemented two versions of the local matching filter described in this paper and tested them on the four benchmark images. These implementations include CPU versions written in MATLAB and a GPU version written in CUDA. The performance numbers reported in this paper were measured on a 2.99-GHz Intel Core 2 Duo processor with 3.25 GB of memory and on a GPU (GeForce 9500GT) with 512 MB of memory. Note that all of the algorithms were run on the same testing platform to achieve a fair comparison.

As demonstrated by the results shown in Table 3, the proposed method is slightly faster than GIF for the testing images both on CPU and GPU platforms. The reason for this is that the total complexity of GIF on three-dimensional color images for disparity maps is  $O(17N)$  [21], while that of the proposed method is  $O(15N)$ , with a tree height  $H = 4$  and therefore a constant  $K = 2^H - 1 = 15$ . Moreover, the proposed method also has the same linear time requirement as the GIF, regardless of the filter kernel size and the intensity range.

Obviously, all the run times increase with the dimensional size of the disparity maps, where the ‘‘Tsukuba,’’ ‘‘Venus,’’ ‘‘Cones,’’ and ‘‘Teddy’’ disparity maps are  $384 \times 288 \times 15$ ,  $434 \times 383 \times 19$ ,  $450 \times 375 \times 59$ , and  $450 \times 375 \times 59$ , respectively. As a result, our CPU implementation processes

a 1-megapixel image in about 16 to 20 seconds, resulting in a time-consuming process. Due to the simple and parallel operations used by our approach, our filter achieves significant performance gains on GPU platform. The total time required for filtering a 1-megapixel image ranges from 0.1 to 0.2 seconds. This represents a speedup from 80 to 200 compared to our CPU implementation.

Consequently, the proposed approach seems to perform slightly better than others in terms of accuracy and computational efficiency.

## 4.2. Influence of Parameter Settings

**4.2.1. Robust Illumination-Independent Behavior.** All of the stereo benchmark images used in Section 4.1 have been acquired under normal lighting conditions and there are no significant variations of luminosity between the two images of a stereo pair. However, this condition is often not valid for a real environment [29, 30]. Due to illumination effects, the color value is not always reliable for stereo matching. Therefore, it has been suggested to supplement the constraint on the gradient in (1), which is invariant to additive illumination changes.

In order to confirm that the proposed method is robust when applied to illumination-variant stereo pairs, PBP results of the altered Tsukuba images with different weight coefficient were presented in Table 4. Refer to Nalpantidis [29], each stereo pair consisted of the left image of the Tsukuba image set and a mount of different versions of the right image whose luminosity alteration ranged from  $-25\%$  to  $+25\%$  with 5% increments.

It can be seen from Table 4 that the algorithm only based on color value (as  $\alpha = 1$ ) leads to many false matches with the lighting nonuniformity, while the quality of the algorithm that just relied on gradient (as  $\alpha = 0$ ) remains almost the same for every tested lighting condition. Moreover the algorithm combining color with gradient value produces the best results for ideal lighting conditions ( $L = 0\%$ ). As a result, the quality

TABLE 4: Evaluation on illumination-variant stereo pairs with different weight coefficient.

$L$	$\alpha$														
	0			0.1			0.5			0.9			1		
	Non	All	Disc	Non	All	Disc	Non	All	Disc	Non	All	Disc	Non	All	Disc
-25%	3.33	4.24	12.27	3.86	4.82	13.50	15.7	16.8	27.2	40.9	41.6	48.1	67.0	67.3	69.9
-20%	3.12	4.00	11.96	3.47	4.36	12.80	17.0	18.1	27.9	40.8	41.5	46.5	60.2	60.6	62.6
-15%	2.97	3.84	11.65	3.27	4.10	11.86	18.8	19.7	29.0	44.8	45.3	46.6	55.4	55.8	55.4
-10%	3.03	3.93	11.87	3.43	4.22	11.23	18.0	18.7	25.3	41.4	41.8	41.9	44.7	45.0	43.7
-5%	3.08	4.07	12.24	3.25	4.01	10.50	12.5	13.3	19.2	25.1	25.7	27.9	26.4	27.0	29.0
0%	<b>3.21</b>	<b>4.21</b>	<b>12.36</b>	<b>2.14</b>	<b>2.94</b>	<b>9.16</b>	<b>1.95</b>	<b>2.75</b>	<b>9.18</b>	<b>2.19</b>	<b>3.09</b>	<b>9.63</b>	<b>2.24</b>	<b>3.15</b>	<b>9.73</b>
5%	3.33	4.32	12.47	2.25	3.09	9.79	10.9	11.7	16.3	21.3	21.9	23.5	22.4	23.0	24.2
10%	3.51	4.54	13.12	3.22	4.10	11.45	19.8	20.5	23.6	41.2	41.4	38.1	42.6	42.8	39.3
15%	3.85	4.88	13.74	3.88	4.82	13.03	20.5	21.3	26.3	49.5	49.9	46.0	52.3	52.6	48.8
20%	4.09	5.14	14.39	4.23	5.24	14.17	19.0	20.1	29.6	54.4	54.9	53.7	59.2	59.5	59.0
25%	4.54	5.62	15.32	4.67	5.76	15.37	20.0	21.1	31.6	53.1	53.7	54.3	62.0	62.5	62.6

TABLE 5: Run time and PBP of the disparity maps vary with respect to tree height increasing.

Height	Time (s)	Non	All	Disc
1	0.031	4.39	5.91	20.63
2	0.056	3.00	3.99	13.78
3	0.128	2.32	3.21	9.93
4	0.270	2.14	2.94	9.16
5	0.499	2.05	2.78	8.90
6	1.203	2.02	2.68	8.99

of our proposed method (when  $\alpha = 0.1$ ) can be less affected by any difference of the lighting conditions and be satisfied with a suitable accuracy.

**4.2.2. Selection of the Tree Height.** The first step is to discuss how tree height affects the performance of the proposed method. ‘‘Tsukuba’’ was chosen as the test image, and the GPU run time and PBP of the disparity maps were recorded with increasing tree height, as shown in Table 5. Note that the spatial  $\sigma_s = 11$  and the range  $\sigma_r = 0.08$  are constants.

It is clear from the second column that the proposed algorithm will increase greatly in compilation time with increasing tree height. Because the number of sampling points  $K = 2^H - 1$  increases with tree height, the greater number of summation operations (8) for the sampling points will be time consuming. On the contrary, the accuracy of the disparity maps for nonoccluded, depth-discontinuous, and overall regions, which is demonstrated in the last three columns, is dramatically improved with increasing height. The reason for this, as mentioned before, is that increasing the number of sampling points reduces the errors between the continuous integration (4) and the discrete summation (5). Figure 4 shows the first three levels of weights (16) for the test image corresponding to the sampling tree (Figure 1). Similar pixels with relatively large weights are shown in white, while black denotes dissimilar pixel areas with very small weights. Moving down this tree, the large weights will be

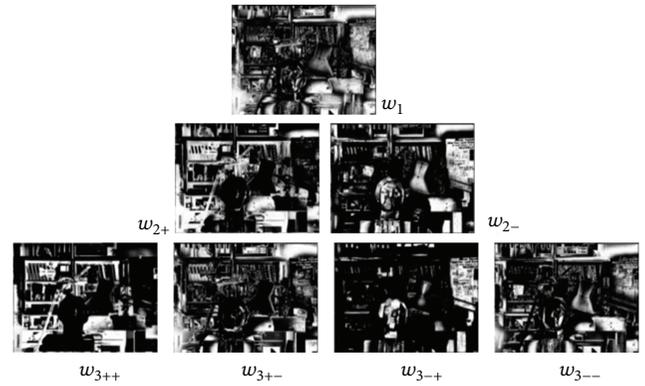


FIGURE 4: The first three levels of weight (16) distribution.

gradually assigned to edge regions, and image integrity will be guaranteed. For example, the missing information from the edge regions of the lamp in  $w_{2-}$ , which is black with very small weights, can be compensated by more detail from the white regions with large weights in  $w_{3++}$  and  $w_{3+-}$ .

However, accuracies improve slightly or even become worse between  $H = 5$  and  $H = 6$  because the spatial and range parameters are constant and unsuitable for the tree height. To confirm this cause,  $H = 6$  is kept constant, and the PBP for ‘‘non’’ are improved, with values of 1.98 and 1.95 when  $\sigma_s = 11$ ,  $\sigma_r = 0.09$  and  $\sigma_s = 12$ ,  $\sigma_r = 0.08$ , respectively. Therefore, the spatial and range parameters also affect performance, which will be further discussed below.

**4.2.3. Influences of  $\sigma_s$  and  $\sigma_r$ .**  $\sigma_s$  and  $\sigma_r$  are two standard deviations used to adjust the spatial similarity and the range similarity, respectively. The spatial spread  $\sigma_s$  is chosen based on the desired amount of low-pass filtering. A large  $\sigma_s$  creates more blurring, meaning that more high-frequency components are removed and the image becomes obviously blurred. Similarly, the range spread  $\sigma_r$  is set to achieve the desired amount of combination of pixel range values. Generally speaking, pixels with range differences less than  $\sigma_r$

TABLE 6: PSNR with different parameters  $\sigma_s$  and  $\sigma_r$ .

$\sigma_r$	$\sigma_s$										
	1	10	20	30	40	50	60	70	80	90	
0.01	11.63	12.99	13.10	13.21	13.24	13.28	13.46	13.38	13.08	12.73	
0.1	12.71	<b>14.01</b>	<b>14.06</b>	<b>14.07</b>	<b>14.01</b>	<b>14.00</b>	<b>14.00</b>	<b>13.86</b>	<b>13.86</b>	<b>13.87</b>	
0.2	12.87	<b>13.99</b>	13.99	13.97	13.89	13.85	13.83	13.67	13.64	13.62	
0.3	12.93	<b>13.96</b>	13.93	13.88	13.77	13.70	13.65	13.47	13.43	13.39	
0.4	12.96	<b>13.94</b>	13.88	13.81	13.70	13.62	13.56	13.38	13.33	13.29	
0.5	12.98	<b>13.93</b>	13.85	13.77	13.64	13.56	13.50	13.32	13.27	13.23	
0.6	12.99	<b>13.92</b>	13.83	13.73	13.60	13.52	13.45	13.27	13.22	13.18	
0.7	12.99	<b>13.91</b>	13.81	13.71	13.57	13.49	13.42	13.24	13.19	13.15	
0.8	13.00	<b>13.91</b>	13.80	13.69	13.55	13.46	13.39	13.21	13.16	13.12	
0.9	13.00	<b>13.91</b>	13.79	13.68	13.53	13.44	13.37	13.19	13.13	13.09	

are mixed together, and those with differences greater than  $\sigma_r$  are removed [13].

The results obtained from varying  $\sigma_s$ ,  $\sigma_r$  in (8) are equivalent to adjusting the spatial and range spread for a bilateral filter. However, the influence of changes in  $\sigma_s$ ,  $\sigma_r$  on the clustering weights (16) is also significant. To analyze the error source qualitatively, the following two propositions are defined.

**Proposition 1.** *More sampling points will be needed for good accuracy when the range spread  $\sigma_r$  is small or the spatial spread  $\sigma_s$  is large. If the height is constant, the matching error would suffer from the edge-losing effect (ELE).*

*Proof.* Using (16), the weight of each pixel is reduced when the value of  $\sigma_r$  is small with respect to the overall range of values in the image or when the set of sampling points  $m$  is dissimilar to the image value  $I$  because  $m$  appears to be hazy due to larger  $\sigma_s$  (9) or (15). Moreover, each  $m$  covers a limited sampling region, which means that in turn, more  $m$  values are needed to adapt to the signal [7].  $\square$

**Proposition 2.** *The filter weights (6) in the proposed method behave more like a low-pass filter when the range spread  $\sigma_r$  is large or the spatial spread  $\sigma_s$  is small. The matching error would be caused by the edge-smoothing effect (ESE).*

*Proof.* Using (16), the weights of all pixels are increased when the value of  $\sigma_r$  is large with respect to the overall range of values in the image or when the set of sampling points  $m$  is similar to the image value  $I$  because  $m$  appears to be less hazy due to smaller  $\sigma_s$  (9) or (15). Therefore, all pixel values in any given neighborhood have approximately the same weight from range filtering for (6), and the resulting filter approximates a standard Gaussian filter [13].  $\square$

“Tsukuba” was chosen as the test image. A fast way to determine the best choice of  $\sigma_s$  and  $\sigma_r$  using the filtering results of peak signal-to-noise ratios (PSNR) [31] is

$$\text{PSNR} = 10 \log_{10} \left\{ \frac{255^2 MN}{\sum_{x,y \in (M,N)} [f(x,y) - g(x,y)]^2} \right\}, \quad (21)$$

where  $M \times N$  is the image size,  $f(\cdot)$  is the local filtering result (as in Figures 3(a)–3(d)), and  $g(\cdot)$  is the ground truth (as in Figure 3(e)). Table 6 shows the PSNR distributions with  $\sigma_s \in (1, 90)$ ,  $\sigma_r \in (0.01, 0.9)$ . The following can be determined.

- (1) The PSNR decreases as  $\sigma_s$  or  $\sigma_r$  becomes smaller when  $\sigma_s \in (1, 10)$  and  $\sigma_r \in (0.01, 0.1)$ . The reason for this is that ESE obeys Proposition 2, that the proposed method behaves more like a low-pass filter when  $\sigma_s$  decreases. The reason for the latter is that ELE obeys Proposition 1 that accuracy is reduced due to lack of more information in the filtering results around the edge regions due to a limited number of sampling points.
- (2) The PSNR decreases with increasing  $\sigma_s$  or  $\sigma_r$  when  $\sigma_s \in (10, 90)$  and  $\sigma_r \in (0.1, 0.9)$ . It obeys Propositions 1 and 2 that the accuracy is reduced due to ELE with large  $\sigma_s$  in a constant-height tree and due to ESE with large  $\sigma_r$  for each sampling point.

From the two findings, it can be confirmed that the optimal values for  $\sigma_s$  and  $\sigma_r$  are approximately 10 and 0.1, respectively, which are shown using bold italic font in Table 6.

The PBP distributions for the “non,” “all,” and “disc” disparity maps were then recorded with  $H = 4$ , but with  $\sigma_s$  and  $\sigma_r$  varying according to  $\sigma_s \in (1, 20)$ ,  $\sigma_r \in (0.01, 0.2)$ , as shown in Figure 5. Results derived from Figure 5 can be summarized as follows.

- (1) All the PBP perform like the results of PSNR; the PBP values increase as  $\sigma_s$  or  $\sigma_r$  becomes smaller. They decrease with increasing  $\sigma_s$  or  $\sigma_r$ , but only up to a certain point, which constitutes the best parameter setting. After that point, the PBP values will gradually increase.
- (2) Figure 5(c) is more obviously different from the first two PBP because it was calculated only from the edge regions. The accuracy reduction refers to the nonoccluded and overall regions generated by ESE or ELE, which are smaller than the depth-discontinuous regions.

Consequently, accuracy was reduced when  $\sigma_s$  and  $\sigma_r$  became too small or too large within a constant-height tree.

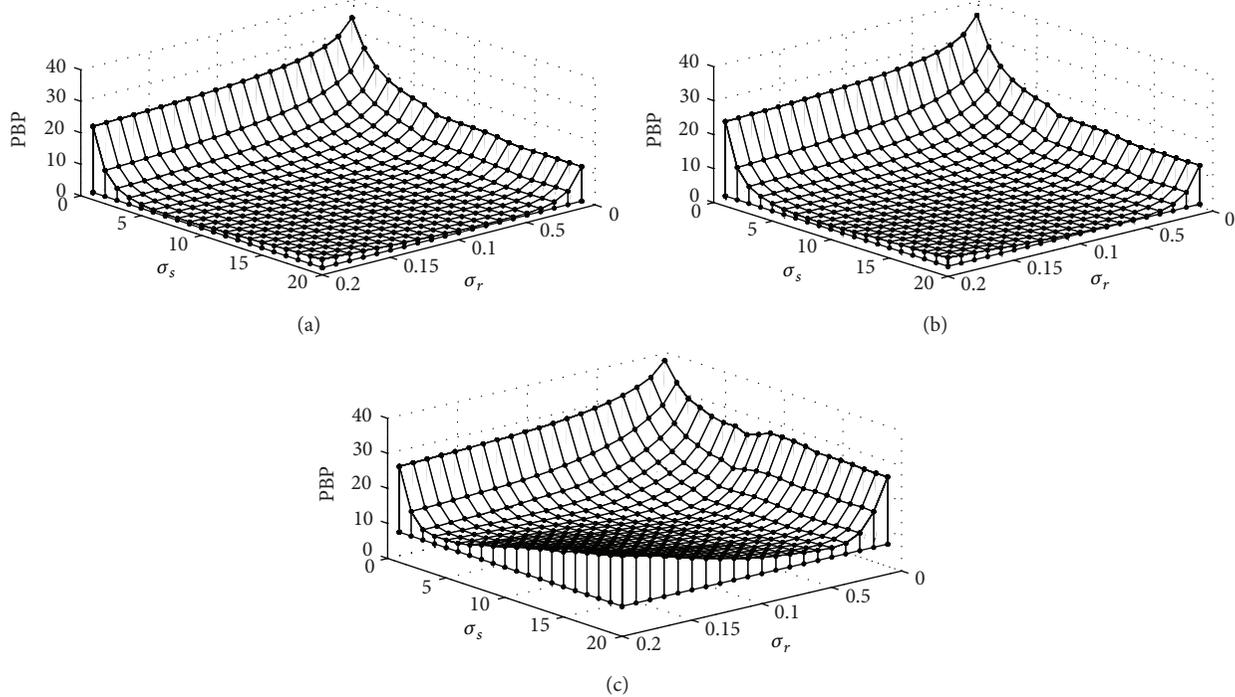


FIGURE 5: PBPP for the (a) “non,” (b) “all,” and (c) “disc” disparity maps with various  $\sigma_s$  and  $\sigma_r$ .

In terms of computational cost, the range component depends linearly  $O(N)$  on the image, regardless of the filter kernel for each sampling layer. To this end, the authors suggest that the tree height be first determined according to the time consumption and then that the filtering results for PSNR be used to determine the general choice of  $\sigma_s$  and  $\sigma_r$ .

## 5. Conclusions and Future Work

In this paper, a new local solution for fast, high-quality dense stereo correspondence has been proposed that focuses on matching cost filtering method which is based on a high-performance hierarchical clustering algorithm. Instead of filtering the matching costs using an edge-preserving smoothing operator as in the popular bilateral filter, the cost aggregation model was adjusted to compute the matching responses for all image pixels at a set of sampling points generated using a clustering method. The computational complexity for this filtering is linear both in the number of image pixels and the number of clustering classes. The experimental results of the comparison have demonstrated that the proposed method outperforms the GIF-based matching algorithm, which is one of the best local methods on the Middlebury benchmark in terms of both speed and accuracy. Moreover, the results of performance tests, which provide effective guidelines for parameter selection, indicate that good accuracy is highly dependent on the weight coefficient, the height of the hierarchical binary tree, and the spatial and range standard deviations. As a result, it can now be confirmed that the proposed approach can be capable of high-speed processing and offer high-quality disparity maps for dense stereo correspondence.

In the experimental results, we show that both of the GI and HC filtering methods make some of the erroneous disparity values due to the lack of texture, which is a traditional challenge for stereo algorithms. The reason is that a pixel's disparity value is obtained by selecting the point of highest matching score and independently of disparity assignments of neighboring pixels. Hence, most of the disparity values in the low-texture areas maybe incorrect using a local matching method. To overcome this bottleneck, the authors plan to make the algorithm capable of handling large untextured regions, which remains an active area for future research [32].

## Acknowledgments

This work was supported by the open project of Beijing Key Laboratory on Measurement and Control of Mechanical and Electrical System (no. KF20121123206), Key Laboratory of Modern Measurement and Control Technology (BISTU), Ministry of Education, Funding Project for Academic Human Resources Development Institutions of Higher Learning under the Jurisdiction of Beijing Municipality (no. PHR201106130), and Funding Project of Beijing Municipal Science & Technology Commission (no. Z121100001612011).

## References

- [1] S. Y. Chen, H. Tong, and C. Cattani, “Markov models for image labeling,” *Mathematical Problems in Engineering*, vol. 2012, Article ID 814356, 18 pages, 2012.
- [2] J. A. Kalomiros, “Dense disparity features for fast stereo vision,” *Journal of Electronic Imaging*, vol. 21, no. 4, Article ID 043023, 2012.

- [3] S. Park and H. Jeong, "High-speed parallel very large scale integration architecture for global stereo matching," *Journal of Electronic Imaging*, vol. 17, no. 1, Article ID 010501, 2008.
- [4] N. Lazaros, G. C. Sirakoulis, and A. Gasteratos, "Review of stereo vision algorithms: from software to hardware," *International Journal of Optomechatronics*, vol. 2, no. 4, pp. 435–462, 2008.
- [5] M. Gong, R. Yang, L. Wang, and M. Gong, "A performance study on different cost aggregation approaches used in real-time stereo matching," *International Journal of Computer Vision*, vol. 75, no. 2, pp. 283–296, 2007.
- [6] K.-J. Yoon and I. S. Kweon, "Adaptive support-weight approach for correspondence search," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 4, pp. 650–656, 2006.
- [7] E. S. L. Gastal and M. M. Oliveira, "Adaptive manifolds for real-time high-dimensional filtering," *ACM Transactions on Graphics*, vol. 31, no. 4, 2012.
- [8] C. Rhemann, A. Hosni, M. Bleyer, M. Bleyer, C. Rother, and M. Gelautz, "Fast cost-volume filtering for visual correspondence and beyond," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 2, pp. 504–511, 2012.
- [9] T. Kanade and M. Okutomi, "Stereo matching algorithm with an adaptive window: theory and experiment," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 9, pp. 920–932, 1994.
- [10] K.-H. Bae, J.-J. Kim, and E.-S. Kim, "New disparity estimation scheme based on adaptive matching windows for intermediate view reconstruction," *Optical Engineering*, vol. 42, no. 6, pp. 1778–1786, 2003.
- [11] S. A. Adhyapak, N. Kehtarnavaz, and M. Nadin, "Stereo matching via selective multiple windows," *Journal of Electronic Imaging*, vol. 16, no. 1, Article ID 013012, 2007.
- [12] S. M. Smith and J. M. Brady, "SUSAN—a new approach to low level image processing," *International Journal of Computer Vision*, vol. 23, no. 1, pp. 45–78, 1997.
- [13] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proceedings of the IEEE 6th International Conference on Computer Vision*, pp. 839–846, January 1998.
- [14] A. Buades, B. Coll, and J.-M. Morel, "A non-local algorithm for image denoising" in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, pp. 60–65, June 2005.
- [15] Y. S. Heo, K. M. Lee, and S. U. Lee, "Simultaneous depth reconstruction and restoration of noisy stereo images using non-local pixel distribution," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '07)*, June 2007.
- [16] F. Porikli, "Constant time  $O(1)$  bilateral filtering," in *Proceedings of the 26th IEEE Conference on Computer Vision and Pattern Recognition (CVPR '08)*, pp. 1–8, Anchorage, Alaska, USA, June 2008.
- [17] Q. Yang, K.-H. Tan, and N. Ahuja, "Real-time  $O(1)$  bilateral filtering," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPR '09)*, pp. 557–564, Miami, Fla, USA, June 2009.
- [18] M.-H. Ju and H.-B. Kang, "Constant time stereo matching," in *Proceedings of the 13th International Machine Vision and Image Processing Conference (IMVIP '09)*, pp. 13–17, Dublin, Republic of Ireland, September 2009.
- [19] J. Chen, S. Paris, and F. Durand, "Real-time edge-aware image processing with the bilateral grid," *ACM Transactions on Graphics*, vol. 26, no. 3, Article ID 1276506, pp. 103–1–103–9, 2007.
- [20] C. Richardt, D. Orr, I. Davies, A. Criminisi, and N. A. Dodgson, "Real-time spatiotemporal stereo matching using the dual-cross-bilateral grid," in *Proceedings of the 11th European Conference on Computer Vision (ECCV '10)*, vol. 6313 of *Lecture Notes in Computer Science*, pp. 510–523, Springer, Heraklion, Greece, 2010.
- [21] K. He, J. Sun, and X. Tang, "Guided image filtering," in *Proceedings of the 11th European Conference on Computer Vision (ECCV '10)*, vol. 6311 of *Lecture Notes in Computer Science*, pp. 1–14, Springer, Heraklion, Greece, 2010.
- [22] L. De-Maeztu, S. Mattoccia, A. Villanueva, and R. Cabeza, "Linear stereo matching," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV '11)*, pp. 1708–1715, Barcelona, Spain, November 2011.
- [23] Q. Yang, "A non-local cost aggregation method for stereo matching," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '12)*, pp. 1402–1409, Providence, RI, USA, 2012.
- [24] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International Journal of Computer Vision*, vol. 47, no. 1-3, pp. 7–42, 2002.
- [25] H. Hirschmüller and D. Scharstein, "Evaluation of cost functions for stereo matching," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '07)*, pp. 1–8, Minneapolis, Minn, USA, June 2007.
- [26] T. Brox and J. Malik, "Large displacement optical flow: descriptor matching in variational motion estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 3, pp. 500–513, 2011.
- [27] "Middlebury stereo vision database," <http://vision.middlebury.edu/stereo/>.
- [28] K. Mühlmann, D. Maier, J. Hesser, and R. Männer, "Calculating dense disparity maps from color stereo images, an efficient implementation," *International Journal of Computer Vision*, vol. 47, no. 1-3, pp. 79–88, 2002.
- [29] L. Nalpantidis and A. Gasteratos, "Stereo vision for robotic applications in the presence of non-ideal lighting conditions," *Image and Vision Computing*, vol. 28, no. 6, pp. 940–951, 2010.
- [30] L. Nalpantidis and A. Gasteratos, "Biologically and psychophysically inspired adaptive support weights algorithm for stereo correspondence," *Robotics and Autonomous Systems*, vol. 58, no. 5, pp. 457–464, 2010.
- [31] K. Bae, J. Ko, and J. Lee, "Stereo image reconstruction using regularized adaptive disparity estimation," *Journal of Electronic Imaging*, vol. 16, no. 1, Article ID 013013, 2007.
- [32] Q. Yang, L. Wang, R. Yang, H. Stewénius, and D. Nistér, "Stereo matching with color-weighted correlation, hierarchical belief propagation, and occlusion handling," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 3, pp. 492–504, 2009.

## Research Article

# Knowledge Mining Based on Environmental Simulation Applied to Wind Farm Power Forecasting

Dongxiao Niu,<sup>1</sup> Ling Ji,<sup>1</sup> Qingguo Ma,<sup>2</sup> and Wei Li<sup>3</sup>

<sup>1</sup> Research Institute of Technology Economics Forecasting and Assessment, School of Economics and Management, North China Electric Power University, Beijing 102206, China

<sup>2</sup> School of Management, Zhejiang University, Hangzhou 310058, China

<sup>3</sup> MOE Key Laboratory of Regional Energy Systems Optimization, S&C Academy of Energy and Environmental Research, North China Electric Power University, Beijing 102206, China

Correspondence should be addressed to Dongxiao Niu; niudongxiao617@126.com

Received 8 June 2013; Accepted 17 August 2013

Academic Editor: Orwa Jaber Housheya

Copyright © 2013 Dongxiao Niu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Considering the inherent variability and uncertainty of wind power generation, in this study, a self-organizing map (SOM) combined with rough set theory clustering technique (RST) is proposed to extract the relative knowledge and to choose the most similar history situation and efficient data for wind power forecasting with numerical weather prediction (NWP). Through integrating the SOM and RST methods to cluster the historical data into several classes, the approach could find the similar days and excavate the hidden rules. According to the data reprocessing, the selected samples will improve the forecast accuracy echo state network (ESN) trained by the class of the forecasting day that is adopted to forecast the wind power output accordingly. The developed methods are applied to a case of power forecasting in a wind farm located in northwest of China with wind power data from April 1, 2008, to May 6, 2009. In order to verify its effectiveness, the performance of the proposed method is compared with the traditional backpropagation neural network (BP). The results demonstrated that knowledge mining led to a promising improvement in the performance for wind farm power forecasting.

## 1. Introduction

With the increasing resources constraints and environment pressure, wind power generation as an important application of renewable energy gains more and more concern. China has enormous potential in the wind energy utilization, with rich wind energy resource mainly in grass land or gobi of northwest, north, and northeast China, as well as coastal area and islands in east and southeast China. Due to the fast developed wind power technology and the distribution characteristics of wind resource, the trend of wind power generation development is concentrated with the large scale. According to statistics, by the end of 2012, the capacity of grid-connected wind generation has been 62660 MW and generated 100.8 billion kWh clean electricity, about 2% of the total electricity supply in China during 2012 [1].

Moreover, in wind resources system, various uncertainties exist in a number of system components as well as their

interrelationships, such as the random characteristics of natural processes (e.g., climate change) and weather conditions, the errors in estimated modeling parameters, and the complexities of system operation. Along with a large amount of wind power integrated into the power system, the intrinsic intermittency and uncertainties of wind resources will cause a great impact on the stability of the whole power system. Thus, precise short-term forecast of wind power farm will be necessary and bring lots of benefits, including cutting large spinning reserve, reducing the cost of wind power generation, and improving the safety and reliability of power system operation.

Previously, the physical method and the statistical approach are the main way for wind power forecasting, and physical conditions such as geography, topography, temperature, and pressure are employed to calculate the power generation capacity of wind farm [2, 3]. These methods require accurate relative meteorological data and difficult

modeling processes. In addition, they usually need to obtain the relationship between abundant historical data of wind power and other variances and are used for long-term wind power forecast, based on a large data base. Therefore, the emphasis of the methods is to build time series or dynamic model based on the experience gained from historical data.

Considering many uncertain factors and disturbances of wind power generating systems, the variation of wind power is largely affected by the complex random factors (e.g., wind speed); significant errors can be easily generated by time series model and regression algorithm, and expert system needs a mass of knowledge and experience with worse maintainability. As a popular simulation method, neural network is an alternative for handling random nonlinear complex mapping without indentifying the transform rules and has been applied in many fields (e.g., environmental modeling and system analysis) [4–6]. In this approach, the historical data of wind power and impact factors (e.g., uncertain parameters) are taken as the input variables. Moreover, the forecasting model is built after learning and training intelligently to avoid the errors or even minimize the impact of random factors on the simulation process. These advantages could lead the neural network to be a suitable method for the complex modeling problem.

In addition, knowledge mining can extract the connotative, unknown, and valuable knowledge or rules from the large-scale database, which is an area of the most extensive application value [7]. Knowledge mining has been widely applied, such as fuzzy time series [8], customer relationship management [9], and agricultural production prediction [10]. Climate factors, for example, wind speed, wind direction, temperature, and humidity, have a great impact on wind farm power changes [11]. Therefore, considering the characters of wind power forecasting, knowledge mining based on environmental simulation is proposed to build a data and knowledge base including history wind power and relative climate factors and to extract the similar historical situations though knowledge mining. Meanwhile, the new sample set would be generated through the historical situations to forecast the wind power with a similar characteristic in future. Moreover, classification is an important topic in data mining research. Given a set of data records, the classification problem is concerned with the discovery of classification rules that can allow records to be correctly classified.

Based on the above mentioned, in this paper, the objective is to propose an improved self-organization mapping neural network to classify the historical data and extract the useful rules for a precise wind power forecasting. A subset of the samples with the similar weather condition is used for training the neural network. The remaining of this paper is organized as follows. Section 2 describes the self-organization mapping based on rough set theory after a brief introduction of relative theory. Detailed structure of echo state network is proposed in Section 3. In Section 4, a case study is carried out to demonstrate the effectiveness of the proposed approach, and the performance is compared with the traditional neural network. The results analysis and discussion are also demonstrated. Finally, some remarks and conclusions of this study are presented in Section 5.

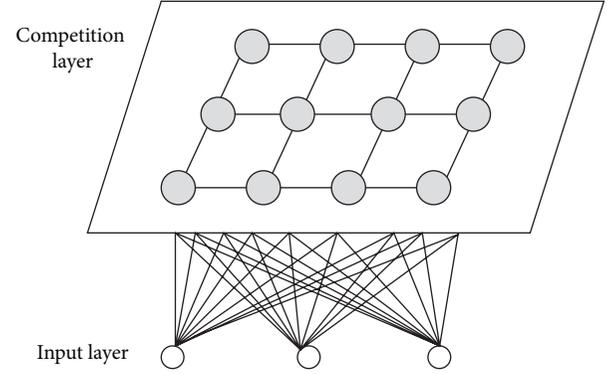


FIGURE 1: The schematic diagram of the SOM neural network.

## 2. Knowledge Mining Based on Environmental Simulation

**2.1. Self-Organization Mapping Theory.** Self-organization mapping (SOM), firstly proposed by Kohonen in 1981, is a neural network with unsupervised learning [12]. It has certain topological structure which is adjusted through the input information, and the pattern recognition is completed by the synergy among multiple neurons [13, 14]. The special idea of SOM is that there is no need to initialize the cluster center or guidance information and that the weight information of neurons is self-adaptively adjusted by input data.

Figure 1 illustrates the schematic diagram of the SOM neural network. Generally, there are two layers in SOM neural network, the input layer and the competition layer. The number of neurons in input layer is equal to the dimension of samples. Every neuron in the input layer connects the neurons in the competitive layer with variable weight values. The neurons in competitive layer will compete for the opportunity to respond to the input pattern, and the weight with the closest match to the presented input pattern is the winner neuron or the best matching unit (BMU). There also exist partial connections between the neurons in the competitive layers. The two-dimension form is the most common form of the neurons' arrangement in competition. The basic principle of SOM is described as follows.

(1) *Competition Process.* Let  $P$  be the input samples with  $m$  dimensions  $P = [p_1, p_2, \dots, p_m]^T$ , where  $m$  is the number of the input neurons. The number of the neurons in competition layer with two dimensions is  $N$  ( $N = n * n$ ). The connection weight between the input layer and the competition layer is denoted as  $W = [W_1^T, W_2^T, \dots, W_N^T]^T$ , where  $W_i^T = [w_{i1}, w_{i2}, \dots, w_{im}]$ .

Calculate the inner product  $n$  of input vector and the connection weight:

$$n = [n_1, n_2, \dots, n_N] = WP = [W_1^T P, W_2^T P, \dots, W_N^T P]^T. \quad (1)$$

Select the winner neuron in the competition process or the best matching unit (BMU). The basic rule is that the larger the inner product, the closer the neuron to the input vector,

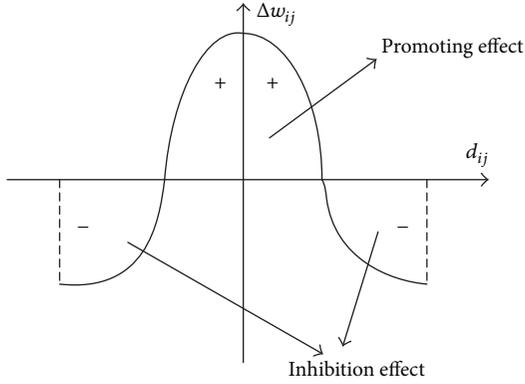


FIGURE 2: The effect of a neuron to its neighbors.

which indicates that the neuron matches to the presented input pattern. The winning formula is as follows:

$$a = \text{compet}(n) = \begin{cases} 1, & i = i^*, \\ 0, & i \neq i^*, \end{cases} \quad (2)$$

where  $n_i^* \geq n_i$ , for all  $i$ , and  $i^* \leq i$ , for all  $n_i = n_i^*$ . Finally, only one neuron wins; thus, the result is as similar as  $a = [0, \dots, 1, \dots, 0]$ , and  $i^*$  is the winner neuron.

(2) *Learning Process.* SOM neural network is arranged according to the two-dimensional structure; each neuron has a promoting effect to the neighboring neurons and on the contrast an inhibition effect to those far away, as shown in Figure 2.

In Figure 2  $d_{ij}$  denotes the distance between the neuron and its neighbor, and  $\Delta w_{ij}$  is the change of the connection weight. It is indicated that the neurons within the certain scope are promoted with weights increasing, while the neurons out of the scope are inhibited, and the weight would be reduced.

In SOM neural network, the weight is adjusted according to Kohonen learning rules. The main idea is to make the winner neuron and its neighbors closer to the input sample by modifying the weights. The formula is expressed as

$$w_i(t) = w_i(t-1) + \alpha(t)(P_i(t) - w_i(t-1)), \quad (3)$$

$$\alpha(t) = \alpha(0) * \left(1 - \frac{t}{T}\right), \quad (4)$$

$$N(t) = N(0) * \left(1 - \frac{t}{T}\right), \quad (5)$$

where  $w_i(t)$  and  $w_i(t-1)$  represent the weight vector of neuron  $i$  at time  $t$  and  $t-1$ , respectively.  $P(t)$  is the input sample at time  $t$ . And  $\alpha(t)$  means the learning rate at time  $t$ , ranging in  $[0, 1]$ . At the beginning, the learning rate is the largest, and the value of  $\alpha(t)$  is decreasing with the training. Since the weight shock of the neurons may occur during the training process, the learning rate need to be reduced gradually. Training neighborhood  $N(t)$  represents the neurons around the winner neuron whose weights will be adjusted. In the initial training, the scope of neighborhood is the slargest,

which provides more neurons learning opportunity. With the increase of training, each neuron represents its own categories, and its neighborhood will be less.

Train the SOM neural network with all the input samples several times; after the procedure mentioned above, the neurons in competitive layer represent each cluster center, which achieves the clustering effect. As a result, the trained network can be utilized for pattern recognition. The algorithm itself can be summarized as follows.

*Step 1.* Initialize the SOM neural network. All the weight values  $w_{ij}$  are initialized to be random in  $[0, 1]$ .

*Step 2.* A training vector  $P_k$  is picked randomly from the training set.

*Step 3.* Calculate the Euclidean distance between input vector  $P_k$  and the weight vector  $W_i$ ;  $d_{ik} = \|P_k - W_i\|$ .  $\min\{d_{ik}\}$  is the winning node and becomes the BMU.

*Step 4.* According to (3), the values of weight vectors for the winning neuron are updated.

*Step 5.* Choose the new input vector and repeat Steps 3 and 4 until stop criterion is satisfied, for example, reaching sufficiently large iterations.

In fact, during the competitive procedure, there may be several neurons that are closely matched with the input vector, thus judging only one of them as the winner seems improper. Considering this issue, the concept of rough set theory is introduced to solve this problem, and self-organized neural network based on rough set is proposed in this paper.

2.2. *Rough Set Theory.* Rough set theory (RST) is a useful tool for data mining and decision support. In particular, it is popular in dealing with the incomplete information, vague concept, and uncertain data [15]. Besides, combined with other data mining algorithm, it can produce more hybrid data mining algorithm [16, 17].

Suppose  $U$  is the nonempty universe with finite members;  $R$  is the equivalence relation in  $U$ ; thus, the knowledge base can be expressed as a relation system  $K = (U, R)$ .

For the subset  $P \subseteq R$  and  $P \neq \emptyset$ , the intersection of all the equivalence relation among  $P$  can be called  $P$ -indiscernibility relation, defined by  $\text{IND}(P)$ :

$$[x]_{\text{IND}(P)} = \bigcap_{R \in P} [x]_R, \quad (6)$$

where  $[x]_R$  represents the equivalence class containing  $x \in U$  in relation  $R$ .

Lower and upper approximations are the important concept of RST. They help to measure the description of uncertain knowledge. Suppose  $X$  is the subset of  $U$ ; then, the lower approximation  $\underline{R}X$  and upper approximation  $\overline{R}X$  are

$$\begin{aligned} \underline{R}X &= \{x \in U : [x]_R \subseteq X\}, \\ \overline{R}X &= \{x \in U : [x]_R \cap X \neq \emptyset\}. \end{aligned} \quad (7)$$

Meanwhile, the boundary region  $BN_R(X) = \overline{RX} - \underline{RX}$ , and it consists of those objects that cannot be classified with certainty as members of  $X$  with the knowledge in  $R$ .

If  $BN_R(X) \neq \phi$ , it indicates that  $\overline{RX} \neq \underline{RX}$ , and  $X$  cannot be expressed by the equivalence class of  $R$  precisely. Thus, the set  $X$  is called ‘‘rough’’ (or ‘‘roughly definable’’); otherwise,  $X$  is crisp.

**2.3. SOM Neural Network Combined with RST.** RST can solve the uncertain or imprecise knowledge expression; thus, it could be employed to deal with the imprecise problem in learning the process of SOM neural network. The novel network still has two-layer structure, while the different of the traditional SOM is in the competitive layer. In the competitive layer, each neuron contains upper approximation and lower approximation. In order to judge which neuron wins, we can determine that the input vector belongs to the lower approximation of a neuron exactly or to the upper approximation of several neurons imprecisely. Through this process, the imprecise problem in judging the winner will be solved properly.

Besides, we can set different learning rates for these two different matching results. If the input vector belongs to the lower approximation, it will get greater learning rate  $\alpha_{low}$ ; otherwise, it will get a lower learning rate  $\alpha_{up}$ . The idea is that when the input vector belongs to a pattern exactly, it can accelerate the learning; when it belongs to a pattern imprecisely, it will reduce its learning effect. The key issue of the novel SOM neural network is how to determine the input vector that belongs to a certain neuron or a set of neurons.

After selecting the best match neuron, it still needs to choose some suboptimum neurons using the suboptimum match degree, calculated as

$$n_{jk} = \frac{n_k}{n_j}, \quad 1 \leq k \leq N, \quad k \neq j, \quad (8)$$

where  $n_{jk}$  is the key factor to determine the lower or upper approximation. Then, it would define the set of suboptimal neurons:

$$V = \{w_k \mid n_{ij} \geq b \text{ and } 1 \leq k \leq N, \text{ and } k \neq j\}, \quad (9)$$

where  $w_k$  is the weight of the  $k$ th neuron;  $b$  is the threshold. Set  $V$  is the collection of the suboptimum neurons with match degree higher than  $b$ . If set  $V$  is empty, it indicates that there are no other close match neurons expect the best match one, and the input vector belongs to its lower approximation. Otherwise, the input vector belongs to the upper approximation of best match and suboptimal neurons. The different learning processes of SOM neural network are expressed, respectively.

- (1) The neuron belongs to the lower approximation exactly:

$$w_i(t) = w_i(t-1) + \alpha_{low}(t)(P_i(t) - w_i(t-1)),$$

$$\alpha_{low}(t) = \alpha_{low}(0) * \left(1 - \frac{t}{T}\right). \quad (10)$$

- (2) The neuron belongs to the upper approximation imprecisely:

$$w_i(t) = w_i(t-1) + \alpha_{up}(t)(P_i(t) - w_i(t-1)),$$

$$\alpha_{up}(t) = \alpha_{up}(0) * \left(1 - \frac{t}{T}\right), \quad (11)$$

where  $T$  is the total training times;  $\alpha_{low}(t)$  and  $\alpha_{up}(t)$  are the learning rates of lower approximation and upper approximation.

The detailed procedure of the proposed SOM neural network is presented as follows.

*Step 1.* Initialize the network, set  $t = 0$ , and  $P(0)$  denotes the input sample vector; initial weight  $w_{ij}(0)$  is a little random number;  $\alpha_{low}(0)$  and  $\alpha_{up}(0)$  are set at 0.9 and 0.5, respectively.

*Step 2.* According to (1), calculate the inner product of the input sample and the neurons in the output layers.

*Step 3.* Select the best match output neuron.

*Step 4.* Select the suboptimal match neurons as a collection  $V$ .

*Step 5.* Adjust the weights according to (10)-(11).

*Step 6.* Turn to Step 2 and repeat the process until all the samples have been tested or the learning rates have been reduced to 0.

### 3. Echo State Network

Recurrent neural networks (RNNs) are very powerful tools for solving complex temporal machine learning tasks [18]. In 2001, a new approach to RNN design and training was respectively proposed under the names of liquid state machines and echo state networks. Its reservoir computing (RC) is an RNN technique that offers a solution to the many problems associated with typical RNN architectures which have prevented their widespread use [19–21].

The classic echo state network contains three layers: input layer, hidden layer, and output layer (shown in Figure 2). The hidden layer is also called dynamic reservoir. Among the traditional circulate network, the scale of neurons is controlled within 12, while there is an abundance of neurons in the reservoir of ESN, about 20 to 500, with good short-term memory. Suppose there are  $L$  units in the input layer,  $M$  units in the output layer, and  $N$  units in the hidden layer. Generally,  $W^{in}$  represents the connection weight matrix of input layer;  $W$  means the connection weight within the reservoir, which keeps 1%~5% sparsely connected. In addition, the spectral radius is usually less than 1. These ensure the reservoir with dynamic memory and certain stability.  $W^{out}$  and  $W^{back}$  denote the connection weight matrix of output layer and feedback. It should be noticed that  $W^{in}$ ,  $W$ , and  $W^{out}$  are decided randomly before the network is established, and once determined they would have not changed.  $W^{out}$  is finally gained by training. Therefore, the main goal of network

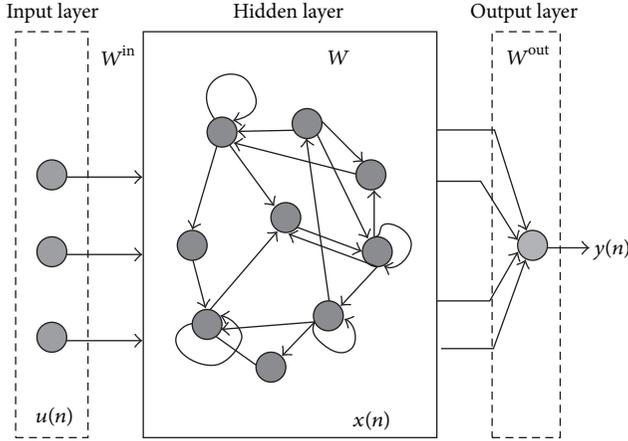


FIGURE 3: The schematic view of echo state network.

training is to determine the value of  $W^{\text{out}}$ . The schematic view of echo state network is presented in Figure 3.

The primary algorithm of echo state network is to inspire the reservoir by input information and to generate the state variables in the reservoir. Though linear regression between the state variables and desire output information, the connection weight of output layer can be determined. The state variables and the output are updated as follows:

$$\begin{aligned} x_{D_i}(n) &= f_{D_i} \left( W_{D_i}^{\text{in}} u(n) + W x_{D_i}(n-1) + W_{D_i}^{\text{back}} y_{D_i}(n-1) \right), \\ \widehat{y}_{D_i}(n) &= f_{D_i}^{\text{out}} \left( W_{D_i}^{\text{out}} \left( u_{D_i}(n), x_{D_i}(n) \right) \right), \\ W_{D_i}^{\text{out}} &= M_{D_i}^{-1} T_{D_i}, \end{aligned} \quad (12)$$

where  $T_{D_i} = [u_{D_i}(n) \ x_{D_i}(n)]$ ,  $M_{D_i} = [y_{D_i}(n)]$ .  $f_{D_i}$  and  $f_{D_i}^{\text{out}}$  are the activation functions of the reservoir and the output, and the most commonly used is a typical hyperbolic tangent function.

#### 4. Case Study and Results Analysis

A wind farm in northwest of China is considered as a case study to demonstrate the effectiveness of the proposed approach. In the study area, there are 66 wind turbines on the wind farm with total capacity 49.5 MW. The historical meteorological data and wind power data from April 1, 2008, to May 6, 2009, are taken as the database. The forecasting model is solved though Matlab on a single core of a 32-bit Lenovo workstation running on Windows7 with 2 dual-core 2.60 GHz CPU and 4.0 GB of RAM. We extract rules from the past information to forecast the wind power load. The main factors considered here are wind scale, temperature, and humidity. The feature vector  $V$  for knowledge mining is described as

$$\begin{aligned} V = & \left( \text{HT}_t, \text{LT}_t, W_t, \text{HT}_{t-1}, \text{LT}_{t-1}, W_{t-1}, \text{MAX } W_{t-1}, \right. \\ & \left. \text{MIN } W_{t-1}, \text{AVER } W_{t-1} \right), \end{aligned} \quad (13)$$

TABLE 1: The cluster results and effectiveness analysis.

Class number	Sample number	SSE
1	118	0.1845
2	48	0.2145
3	39	0.0954
4	30	0.1547
5	26	0.3458
6	75	0.5476
7	64	0.3547

where  $\text{HT}_t$  and  $\text{LT}_t$  are the highest and lowest temperatures on the  $t$ th day;  $\text{HT}_{t-1}$  and  $\text{LT}_{t-1}$  are the highest and lowest temperatures on the  $t-1$ th day;  $W_t$  and  $W_{t-1}$  are the wind scales on the  $t-1$ th day;  $\text{MAX } W_{t-1}$ ,  $\text{MIN } W_{t-1}$ , and  $\text{AVER } W_{t-1}$  are the maximum, minimum, and average wind speeds on the  $t-1$ th day.

In order to eliminate the dimension influence among different variables, data preprocessing is the first job need to be done.

The proposed RS-SOM neural network is employed to cluster the history information. To evaluate the compactness of the clustering results, the sum of squared error (SSE) is adopted in this study. The smaller the SSE, the better the effect. And it is calculated as

$$\begin{aligned} \text{SSE} &= \sum_k^K \sum_{x_i \in C_k}^n \|x_i - \mu_k\|^2, \\ \mu_{k,j} &= \frac{1}{N_k} \sum_{x_i \in C_k}^n x_{ij}, \end{aligned} \quad (14)$$

where  $C_k$  is the set of each cluster,  $\mu_k$  is the mean of the  $k$ th cluster, and  $N_k$  is the number of samples belonging to the  $k$ th cluster.

The cluster results are shown in Table 1. It can be seen that the 400 samples in the data base are divided into 7 classes. The SSE of class 3 is the smallest, while that of class 6 is the largest. This indicated that the samples in class 3 are the most closely similar and the shape of the curve can almost reflect the fluctuation of the wind output power on those days. Besides the average wind power output curve of each class is illustrated in Figure 4. Each curve has significant features and is obviously different from the others. For example, in class 1, the valley of most samples' power output is at about 10:00, while the peak time is at nearly 23:00. This is the most common situation with the largest number of class member 118. The class with the least samples is class 5, whose shape is extremely irregular.

Since the weather report may not be accurate, and the more the time lag, the worse the forecasting result, and this study only forecasted the wind power output on the next two days accordingly. The new input pattern can be discriminated using the trained RS-SOM neural network. The samples in the same class with the forecasting day are selected to train the ESN network for the forecasting day. In order to test the performance of ESN, BP neural network

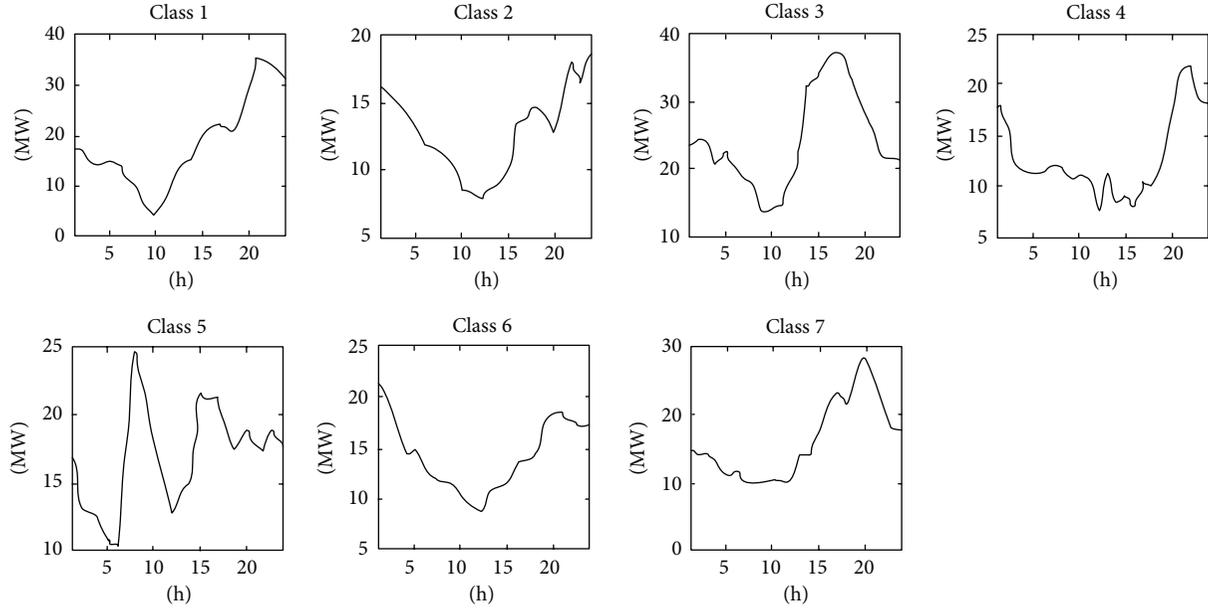


FIGURE 4: The wind farm output curves of different classes.

which has been widely used in load forecasting is also applied for the same task. Commonly used error evaluation indexes, including mean absolute percentage error (MAPE), mean absolute error (MAE), root mean square error (RMSE), and normal root mean square error (NRMSE), are used to discuss the performance of different forecasting methods:

$$\begin{aligned} \text{MAPE} &= \frac{1}{N} \sum_{i=1}^N \left| \frac{y_i - \hat{y}_i}{y_i} \right|, \\ \text{MAE} &= \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i|, \\ \text{RMSE} &= \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2}, \\ \text{NRMSE} &= \sqrt{\frac{\sum (\hat{y}_i - y_i)^2}{\sum y_i^2}}, \end{aligned} \quad (15)$$

where  $\hat{y}_i$  is the forecasting value,  $y_i$  is the actual value, and  $N$  is the number of samples.

Figure 5 shows the forecasting results of different methods and the actual wind power output. It indicates that the overall trend of forecasting power is in accordance with the actual situation. However, as the forecasting of peak and valley, the performance of ESN is obviously greater than BP model. The deviation of the latter is larger at the extremism values. During the period of 1:00 to 8:00, the performance of both methods is well, while, at night from 18:00 to 24:00, the deviation is larger. The error evaluation indexes for both methods are presented in Table 2, which shows that all of the indexes of ESN are lower than BP. The MAPE and MAE of ESN are 0.1366 and 1.7771 MW, respectively, lower than those

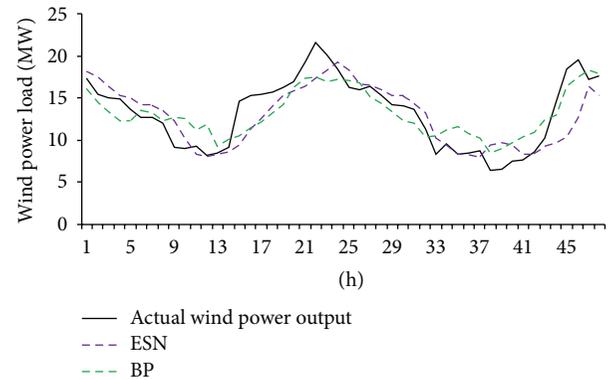


FIGURE 5: Forecasting results of different models and the actual wind power output.

TABLE 2: Error evaluation of different forecasting models.

Methods	MAPE	MAE (MW)	RMSE	NRMSE
ESN	0.1366	1.7771	2.2171	0.0019
BP	0.1709	1.9607	2.4320	0.0023

of BP 0.1709 and 1.9607 MW. The RMSE and NRMSE of ESN forecasting are 2.2171 and 0.0019, and those of BP are 2.4320 and 0.0023, respectively. As mentioned, the forecast accuracy for the study area obtained from ESN model is substantially more than that typically seen from the BP model. This extensive comparison reflects forecasts for all available 66 wind turbines. It is illustrated that the accuracy of the ESN forecasts was consistently better for the year and for each wind plant. It should be noted that accuracy would be even better if the data were adjusted for curtailments.

Compared with the results obtained from the two methods, it is indicated that the reason why accurate wind power forecasting of ESN is more than that of the BP model and the BP model has a higher degree of error than the ESN method is that (1) the wind farm is located in the northwest of China, the management system of meteorological measurements is imperfect, and meteorological data at the wind plant often are of poor quality and contribute to inaccuracy in traditional forecasting approaches; (2) wind direction readings from standard met towers might not even be applicable since surrounding terrain can affect this movement, and wind can ramp up or down quickly, that could lead to misleading information. All of the above problems with a very complex and important prediction process could directly affect the accuracy of the traditional forecasts. Data management also is vital to the accuracy of the forecasting method. Through integrating the SOM and RST methods to cluster the historical data in to several classes, the approach could find the similar days and excavate the hidden rules for increasing the forecast accuracy. In addition, it could reflect the uncertainties of the input parameters, avoid the data error, and provide valid data for ESN forecasting.

However, compared with other approaches, there is still much space for improvement of the proposed method. For example, there is no uniform way to determine the relative parameters (as the spectral radius of the connection weight within the reservoir) of the ESN network, which are mainly gained through massive experiments. Besides, other neural networks with universal approximation capability, for example, the radial basis function (RBF) network, will be studied and reformed further to improve the forecasting accuracy and calculation speed.

## 5. Conclusion

Wind power forecasting is an important tool for managing the inherent variability and uncertainty in wind power generation. Increasing the accuracy of forecasting can help to reduce the likelihood of an unexpected gap between scheduled and actual wind power generation, which can be extremely helpful for operators of power systems and wind power plants. In this study, we developed a database by using historical meteorological environment and power output data. Self-organizing map combining rough set theory as a knowledge mining technology is employed to discover and extract the rules. The classified samples are taken as the input of echo state network to train the structure of the network, respectively. Through integrating the SOM and RST methods to cluster the historical data in to several classes, the approach would provide valid data for ESN forecasting. The developed methods are applied to a case of power forecasting in a wind farm located in northwest of China with a wind power data from April 1, 2008, to May 6, 2009. The results demonstrated the successful use of the proposed method, which performs better than BP. The accuracy of prediction has been improved. However, the database is static in this study, which means the information of new samples will not be added into the knowledge pool automatically. And if

the database is small, it may not cover comprehensive situation. Thus, it will be proper for the wind farm with long operation period.

## Acknowledgments

This work was supported in part by the NSFC under Grant no. 71071052 and Grant no. 71201057, as well as “the Fundamental Research Funds for the Central Universities” under Grant no. 12QX23 and Grant no. 13X20. The authors would like to thank the anonymous reviewers and editors for their valuable comments, which greatly helped them to clarify and improve the contents of the paper.

## References

- [1] [http://www.gov.cn/gzdt/2013-04/09/content\\_2373121.htm](http://www.gov.cn/gzdt/2013-04/09/content_2373121.htm).
- [2] Y. Jiang, Z. Song, and A. Kusiak, “Very short-term wind speed forecasting with Bayesian structural break model,” *Renewable Energy*, vol. 50, pp. 637–647, 2013.
- [3] Y. H. Ma, N. B. Wang, F. C. Liu, G. T. Liu, and J. Wang, “A wind power forecast system for Jiuquan wind power base in Gansu province,” *Automation of Electric Power Systems*, vol. 33, no. 16, pp. 88–90, 2009.
- [4] F. Cassola and M. Burlando, “Wind speed and wind energy forecast through Kalman filter of Numerical Weather Prediction model output,” *Applied Energy*, vol. 99, pp. 154–166, 2012.
- [5] U. H. Ashraf, M. Paras, E. K. Mary, J. Meng, L. C. Chang, and T. Senjyu, “A new strategy for predicting short-term wind speed using soft computing models,” *Renewable and Sustainable Energy Reviews*, vol. 16, no. 7, pp. 4563–4573, 2012.
- [6] C. N. Yao, X. K. Gao, and Y. C. Yu, “Wind speed forecasting by wavelet neural networks: a comparative study,” *Mathematical Problems in Engineering*, vol. 2013, Article ID 395815, 7 pages, 2013.
- [7] X. Y. Pan and L. C. Jiao, “A granular agent evolutionary algorithm for classification,” *Applied Soft Computing*, vol. 11, no. 3, pp. 3093–3105, 2011.
- [8] T. K. Liu, Y. P. Chen, and J. H. Chou, “Extracting fuzzy relations in fuzzy time series model based on approximation concepts,” *Expert Systems with Applications*, vol. 38, no. 9, pp. 11624–11629, 2011.
- [9] E. W. T. Ngai, L. Xiu, and D. C. K. Chau, “Application of data mining techniques in customer relationship management: a literature review and classification,” *Expert Systems with Applications*, vol. 36, no. 2, part 2, pp. 2592–2602, 2009.
- [10] Y. L. Everingham, C. W. Smyth, and N. G. Inman-Bamber, “Ensemble data mining approaches to forecast regional sugarcane crop production,” *Agricultural and Forest Meteorology*, vol. 149, no. 3-4, pp. 689–696, 2009.
- [11] Z. H. Guo, Y. Dong, J. Z. Wang, and H. Y. Lu, “The forecasting procedure for long-term wind speed in the Zhangye area,” *Mathematical Problems in Engineering*, vol. 2010, Article ID 684742, 17 pages, 2010.
- [12] Z. S. Wang, S. R. Bian, and Y. Liu, “The load characteristics classification and synthesis of substations in large area power grid,” *International Journal of Electrical Power & Energy System*, vol. 48, pp. 71–82, 2013.

- [13] L. Hernandez, C. Baladron, and J. M. Aguiar, "Classification and cluster of electricity demand patterns in industrial parks," *Energies*, vol. 5, no. 12, pp. 5215–5228, 2012.
- [14] V. Nourani, A. H. Baghanam, and J. Adamowski, "Using self-organizing maps and wavelet transforms for space-time pre-processing of satellite precipitation and runoff data in neural network based rainfall-runoff modeling," *Journal of Hydrology*, vol. 476, pp. 228–243, 2013.
- [15] S. I. Omurca, "An intelligent supplier evaluation, selection and development system," *Applied Soft Computing*, vol. 13, no. 1, pp. 690–697, 2013.
- [16] P. Georg, W. Richard, and N. Rene, "Dynamic rough clustering and its application," *Applied Soft Computing*, vol. 12, no. 10, pp. 3193–3207, 2012.
- [17] A. Ganivada, S. S. Ray, and S. K. Pal, "Fuzzy rough granular self-organizing map and fuzzy rough entropy," *Theoretical Computer Science*, vol. 466, pp. 37–63, 2012.
- [18] A. A. Ferreira, T. B. Ludermit, and R. B. A. Ronaldo, "An approach to reservoir computing design and training," *Expert Systems with Applications*, vol. 40, no. 10, pp. 4172–4182, 2013.
- [19] J. B. Butcher, D. Verstraeten, B. Schrauwen, C. R. Day, and P. W. Haycock, "Reservoir computing and extreme learning machines for non-linear time series data analysis," *Neural Networks*, vol. 38, pp. 76–89, 2013.
- [20] L. Ji, D. X. Niu, and H. Wu, "Daily peak load forecasting based on Bayesian framework and echo state network," *Power System Technology*, vol. 36, no. 11, pp. 82–86, 2012 (Chinese).
- [21] D. X. Niu, L. Ji, Y. L. Wang, and D. Liu, "Optimization of echo state network with wavelet analysis applied in short-term load forecasting," *Kybernets*, vol. 41, no. 10, pp. 1557–1570, 2012.

## Research Article

# Identification of Code-Switched Sentences and Words Using Language Modeling Approaches

Liang-Chih Yu,<sup>1</sup> Wei-Cheng He,<sup>1</sup> Wei-Nan Chien,<sup>1</sup> and Yuen-Hsien Tseng<sup>2</sup>

<sup>1</sup> Department of Information Management, Yuan Ze University, 135 Yuan-Tung Road, Chung-Li 32003, Taiwan

<sup>2</sup> Information Technology Center, National Taiwan Normal University, 162 Heping East Road, Section 1, Taipei 10610, Taiwan

Correspondence should be addressed to Liang-Chih Yu; [lcyu@saturn.yzu.edu.tw](mailto:lcyu@saturn.yzu.edu.tw)

Received 3 May 2013; Revised 22 July 2013; Accepted 19 August 2013

Academic Editor: Orwa Jaber Housheya

Copyright © 2013 Liang-Chih Yu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Globalization and multilingualism contribute to code-switching—the phenomenon in which speakers produce utterances containing words or expressions from a second language. Processing code-switched sentences is a significant challenge for multilingual intelligent systems. This study proposes a language modeling approach to the problem of code-switching language processing, dividing the problem into two subtasks: the detection of code-switched sentences and the identification of code-switched words in sentences. A code-switched sentence is detected on the basis of whether it contains words or phrases from another language. Once the code-switched sentences are identified, the positions of the code-switched words in the sentences are then identified. Experimental results show that the language modeling approach achieved an  $F$ -measure of 80.43% and an accuracy of 79.01% for detecting Mandarin-Taiwanese code-switched sentences. For the identification of code-switched words, the word-based and POS-based models, respectively, achieved  $F$ -measures of 41.09% and 53.08%.

## 1. Introduction

Increasing globalism and multilingualism has significantly increased demand for multilingual services in current intelligent systems [1]. For example, an intelligent traveling system which supports multiple language inputs and outputs can assist travelers in booking hotels, ordering in restaurants, and navigating attractions. Multinational corporations would benefit from developing automatic multilingual call centers to address customer problems worldwide. In such multilingual environments, an input sentence may contain constituents from two or more languages, a phenomenon known as code-switching or language mixing [2–6]. Table 1 lists several definitions of code-switching described in previous studies.

A code-switched sentence consists of a primary language and a secondary language, and the secondary language is usually manifested in the form of short expressions, such as words and phrases. This phenomenon is increasingly common, with multilingual speakers often freely moving from their native dialect to subsidiary dialects to entirely foreign languages, and patterns of code-switching vary dynamically

with different audiences in different situations. When dealing with code-switched input, intelligent systems such as dialog systems must be capable of identifying the various languages and recognize the speaker's intention embedded in the input [7, 8]. However, it is a significant challenge for intelligent systems to deal with multiple languages and unknown words from various languages.

In Taiwan, while Mandarin is the official language, Taiwanese and Hakka are used as a primary language by more than 75% and 10% of the population, respectively [9]. Moreover, English is the most popular foreign language and compulsory English instruction begins in elementary school. The constant mix of these languages result in various kinds of code-switching, such as Mandarin sentences mixed with words and phrases from Taiwanese, Hakka, and English. Such code-switching is not limited to everyday conversation but can frequently be heard on television dramas and even current events commentary programs. This paper takes a linguistic view towards the problem of code-switching language processing, focusing on code-switching between Mandarin and Taiwanese. We propose a language modeling approach

TABLE 1: Definitions of code-switching.

Study	Definition
Hymes et al. [2]	A common term for alternative use of two or more languages, varieties of a language, or even speech styles
Hoffmann [3]	The alternate use of two languages or linguistic varieties within the same utterance or during the same conversation
Myers-Scotton [4]	The use of two or more languages in the same conversation, usually within the same conversational turn or even within the same sentence of that turn

which divides the problem into two subtasks: the detection of code-switched sentences followed by identification of code-switched words within the sentences. The first step detects whether or not a given Mandarin sentence contains Taiwanese words. Once a code-switched sentence is identified, the positions of the code-switched words are then identified within the sentence. These code-switched words can be used for lexicon augmentation to improve understanding of code-switched sentences.

The rest of this work is organized as follows. Section 2 presents related work. Section 3 describes the language modeling approach to the identification of code-switched sentences and words in the sentences. Section 4 summarizes the experimental results. Conclusions are finally drawn in Section 5, along with recommendations for future research.

## 2. Related Work

Research on code-switching speech processing mainly focuses on speech recognition [9–14], language identification [15, 16], text-to-speech synthesis [17], and code-switching speech database creation [18]. Lyu et al. proposed a three-step data-driven phone clustering method to train an acoustic model for Mandarin, Taiwanese, and Hakka [9]. They also discussed the issue of training with unbalanced data. Wu et al. proposed an approach to segmenting and identifying mixed-language speech utterances [10]. They first segmented the input speech utterance into a sequence of language-dependent segments using acoustic features. The language-specific features were then integrated in the identification process. Chan et al. developed a Cantonese-English mixed-language speech recognition system, including acoustic modeling, language modeling, and language identification algorithms [11]. Hong et al. developed a Mandarin-English mixed-language speech recognition system in resource-constrained environments, which can be realized in embedded systems such as personal digital assistants (PDAs) [12]. Ahmed and Tan proposed a two-pass code-switching speech recognition framework: automatic speech recognition and rescoring [13]. Vu et al. recently developed a speech recognition system for code-switching in conversational speech [14]. For language identification, Lyu et al. proposed a word-based lexical model integrating acoustic, phonetic, and lexical cues to build a language identification system [15]. Yeong and Tan

proposed the use of morphological structures and sequence of the syllable for language identification from Malay-English code-switching sentences [16]. For speech synthesis, Qian et al. developed a text-to-speech system that can generate Mandarin-English mixed-language utterances [17].

Research on code-switching and multilingual language processing included applications of text mining [19–22], information retrieval [23–25], ontology-based knowledge management [26], and unknown word extraction [27]. For text mining, Seki et al. extracted opinion holders for discriminating opinions that are viewed from different perspectives (author and authority) in both Japanese and English [19]. Yang et al. used self-organizing maps to cluster multilingual documents [20]. A multilingual Web directory was then constructed to facilitate multilingual Web navigation. Zhang et al. addressed the problem of multilingual sentence categorization and novelty mining on English, Malay, and Chinese sentences [21]. They proposed to first categorize similar sentences and then identify new information from them. De Pablo-Sánchez et al. devised a bootstrapping algorithm to acquire named entities and linguistic patterns from English and Spanish news corpora [22]. This lightly supervised method can acquire useful information from unannotated corpora using a small set of seeds provided by human experts. For information retrieval, Gey et al. pointed out several directions for cross-lingual information retrieval (CLIR) research [23]. Tsai et al. used the FRank ranking algorithm to build a merge model for multilingual information retrieval [24]. Jung discovered useful multilingual tags annotated in social texts [25]. He then used these tags for query expansion to allow users to query in one language but obtain additional information in another language. For other application domains, Segev and Gal proposed an ontology-based knowledge management model to enhance portability and reduce costs in multilingual information systems deployment [26]. Wu et al. proposed the use of mutual information and entropy to extract unknown words from code-switched sentences [27].

## 3. Language Modeling Approach

Language modeling approaches have been successfully used in many applications, such as grammar error correction [28], code-switching language processing [29], and lexical substitution [30–32]. For our task, a code-switched sentence generally has a higher probability of being found in a code-switching language model than in a noncode-switching one. Thus, we built code-switching and noncode-switching language models to compare their respective probabilities of identifying code-switched sentences and code-switched words within the sentences. Figure 1 shows the system framework. First, a corpus of code-switched and noncode-switched sentences is collected to build the respective code-switching and noncode-switching language models. To identify code-switched sentences, we compare the probability of each test sentence output by the code-switching language model against the output of the noncode-switching one to determine whether or not the test sentence is code-switched. To identify code-switched words within the sentences,

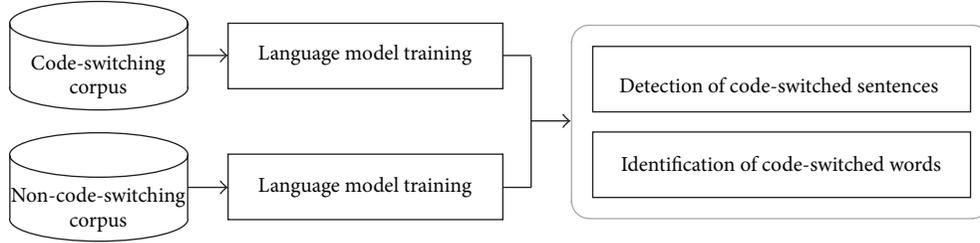


FIGURE 1: Framework of identification of code-switched sentences and words in the sentences.

we select the  $n$ -gram with the highest probability output by the code-switching language model and then compare it against the output of the noncode-switching one to verify whether the  $n$ th word in the given sentence is a code-switched word.

**3.1. Corpus Collection.** A noncode-switching corpus refers to a set of sentences containing just one language. Because Mandarin is the primary language in this study, we used the Sinica corpus released by the Association for Computational Linguistics and Chinese Language Processing (ACLCLP) as the noncode-switching corpus. A code-switching corpus refers to a set of Mandarin sentences featuring Taiwanese words. However, it can be difficult to collect a large number of such sentences, and training a language model on insufficient data may incur the data sparseness problem. Therefore, we used more common Mandarin-English sentences as the code-switching corpus, based on the assumption that the code-switching phenomenon in Mandarin-English sentences has a certain degree of similarity to Mandarin-Taiwanese sentences, because in Taiwan, both English and Taiwanese are secondary languages with respect to Mandarin. The Mandarin-English sentences were collected from a large corpus of web-based news articles, which were then segmented using the CKIP word segmentation system developed by the Academia Sinica, Taiwan (<http://ckipsvr.iis.sinica.edu.tw/>) [33, 34]. The sentences containing words with the part-of-speech (POS) tag “FW” (i.e., foreign word) were selected as code-switched sentences.

**3.2. Detection of Code-Switched Sentences.** Generally, an  $n$ -gram language model is used to predict the  $n$ th word based on the previous  $n - 1$  words using a probability function  $P(w_n | w_1 \cdots w_{n-1})$ . Given a sentence  $S = w_1 \cdots w_k$ , the noncode-switching  $n$ -gram language model is defined as

$$\begin{aligned}
 P_{\overline{cs}}(S) &= P(w_1)P(w_2 | w_1) \cdots P(w_k | w_1 \cdots w_{k-1}) \\
 &= \prod_{i=1}^k P(w_i | w_1 \cdots w_{i-1}) \\
 &\approx \prod_{i=1}^k P(w_i | w_{i-1} \cdots w_{i-n+1}),
 \end{aligned} \tag{1}$$

where  $P(w_i | w_{i-1} \cdots w_{i-n+1})$  is estimated by

$$P(w_i | w_{i-1} \cdots w_{i-n+1}) = \frac{C(w_i \cdots w_{i-n+1})}{C(w_{i-1} \cdots w_{i-n+1})}, \tag{2}$$

where  $C(\cdot)$  denotes the frequency counts of the  $n$ -grams retrieved from the noncode-switching corpus (i.e., Sinica corpus). Instead of estimating the surface form of the next word, the code-switching  $n$ -gram language model estimates the probability that the next word is a code-switched word, that is,  $P(cs_n | w_1 \cdots w_{n-1})$ , defined as

$$\begin{aligned}
 P_{cs}(S) &= P(w_1)P(cs_2 | w_1) \cdots P(cs_k | w_1 \cdots w_{k-1}) \\
 &= \prod_{i=1}^k P(cs_i | w_1 \cdots w_{i-1}) \\
 &\approx \prod_{i=1}^k P(cs_i | w_{i-1} \cdots w_{i-n+1}),
 \end{aligned} \tag{3}$$

where  $P(w_i | w_{i-1} \cdots w_{i-n+1})$  is estimated by

$$P(cs_i | w_{i-1} \cdots w_{i-n+1}) = \frac{C(cs_i \cdots w_{i-n+1})}{C(cs_{i-1} \cdots w_{i-n+1})}. \tag{4}$$

To estimate  $P(cs_n | w_1 \cdots w_{n-1})$ , the code-switching corpus is processed by replacing the code-switched words (i.e., the words with the POS tag “FW”) in the Mandarin-English sentences with a special character  $cs$ . The frequency counts of  $C(cs_i \cdots w_{i-n+1})$  can then be retrieved from the code-switching corpus. This processing may also reduce the effect of the data sparseness problem in language model training.

Once the two language models are built, they can be compared to detect whether a given sentence contains code-switching. That is,

$$c = \frac{P_{cs}(S)}{P_{\overline{cs}}(S)}. \tag{5}$$

The sentence  $S$  is predicted to be a code-switched sentence if the probability of the sentence output by the code-switching language model is greater than that output by the noncode-switching one (i.e.,  $c \geq 1$ ).

**3.3. Identification of Code-Switched Words.** This step identifies the positions of the code-switched words within the sentences. To this end, the code-switching  $n$ -gram language model (3) is applied to each test sentence and the probability of being a code-switched word is assigned to every next word (position) in the sentence. Among all the  $n$ -grams in the sentence, the one with the highest probability indicates the most likely position of a code-switched word. That is,

$$cs^* = \underset{i}{\operatorname{argmax}} P(cs_i | w_{i-1} \cdots w_{i-n+1}), \tag{6}$$

where  $cs^*$  denotes the best hypothesis of the code-switched word in the sentence. However, not all  $n$ -grams with the highest probability suggest correct positions. Therefore, we further propose a verification mechanism to determine whether to accept the best hypothesis. That is,

$$cs = \begin{cases} cs^* & P^*(cs_i | w_{i-1} \cdots w_{i-n+1}) \\ & \geq P(w_i | w_{i-1} \cdots w_{i-n+1}), \\ \phi & P^*(cs_i | w_{i-1} \cdots w_{i-n+1}) \\ & < P(w_i | w_{i-1} \cdots w_{i-n+1}), \end{cases} \quad (7)$$

where  $P^*(cs_i | w_{i-1} \cdots w_{i-n+1})$  represents the probability of the best hypothesis in the code-switching corpus and  $P(w_i | w_{i-1} \cdots w_{i-n+1})$  represents its probability in the noncode-switching corpus. The best hypothesis  $cs^*$  is accepted if its probability in the code-switching corpus is greater than that in the noncode-switching corpus.

## 4. Experimental Results

This section first explains the experimental setup, including experiment data, implementation of language modeling, and evaluation metrics. We then present experimental results for the identification of both Mandarin-Taiwanese and Mandarin-English code-switched sentences and words within the sentences.

**4.1. Experimental Setup.** The test set included 393 sentences of which 131 were Mandarin only (i.e., noncode-switched), while another 131 were Mandarin sentences containing Taiwanese words, and the remaining 131 were Mandarin sentences containing English words. For the evaluation of Mandarin-Taiwanese sentences,  $n$ -gram models for both code-switching and noncode-switching were trained using the SRILM toolkit [35] with  $n = 2$  and 3 (i.e., bigram and trigram). For the evaluation of Mandarin-English sentences, the CKIP word segmentation system [33, 34] was used because it can associate a POS tag “FW” to English words/characters within the sentences. The evaluations metrics included recall, precision,  $F$ -measure, and accuracy. The recall was defined as the number of code-switched sentences correctly identified by the method divided by the total number of code-switched sentences in the test set. The precision was defined as the number of code-switched sentences correctly identified by the method divided by the number of code-switched sentences identified by the method. The  $F$ -measure was defined as  $(2 \times recall \times precision)/(recall + precision)$ . The accuracy was defined as the number of sentences correctly identified by the method divided by the total number of sentences in the test set.

### 4.2. Results

**4.2.1. Evaluation on Mandarin-Taiwanese Code-Switched Sentences.** To identify Mandarin-Taiwanese code-switched sentences, the code-switching and noncode-switching bigram/trigram language models were used to determine whether

TABLE 2: Results of the identification of Mandarin-Taiwanese code-switched sentence.

Methods	Recall	Precision	$F$ -measure	Accuracy
Bi-gram	86.26%	75.33%	80.43%	79.01%
Tri-gram	77.86%	62.20%	69.15%	65.27%

TABLE 3: Results of code-switched word identification in Mandarin-Taiwanese code-switched sentences.

Methods		Recall	Precision	$F$ -measure
Random	Top 1	17.65%	18.32%	17.98%
	Top 2	31.62%	16.41%	21.61%
	Top 3	49.26%	17.05%	25.33%
Word bigram	Top 1	40.46%	41.73%	41.09%
	Top 2	60.31%	31.85%	41.69%
	Top 3	74.81%	27.53%	40.25%
Word trigram	Top 1	14.50%	15.83%	15.14%
	Top 2	35.88%	20.35%	25.97%
	Top 3	55.73%	22.53%	32.09%
POS bigram	Top 1	42.75%	43.08%	42.91%
	Top 2	67.94%	35.18%	46.35%
	Top 3	82.44%	29.35%	43.29%
POS trigram	Top 1	52.67%	53.49%	53.08%
	Top 2	73.28%	39.34%	51.20%
	Top 3	83.97%	31.98%	46.32%

or not each test sentence features code-switching (5), with results presented in Table 2. The bigram language model correctly identified 113 code-switched sentences and 94 noncode-switched sentences, thus yielding 86.26% (113/131) recall, 75.33% (113/150) precision, 80.43%  $F$ -measure, and 79.01% (207/262) accuracy. The trigram language model, however, did not outperform the bigram model, possibly due to the data sparseness problem caused by a lack of sufficient training data for building the trigram language model.

To identify code-switched words in Mandarin-Taiwanese code-switched sentences, all word bigrams and trigrams in each test sentence were first ranked according to their probabilities. The top  $N$  word bigrams/trigrams were then selected as candidates for further verification using (7). For instance, top 1 means that the bigram/trigram with the highest probability in a given test sentence is considered a candidate. If the candidate  $n$ -gram is accepted by the verification method, then the position indicated by the  $n$ -gram will be considered a foreign word. Similarly, top 2 means that the method can propose two candidates for verification. To examine the effect of the data sparseness problem, we used the part-of-speech (POS) tags of words to build additional POS bigram/trigram models from the code-switching corpus. In addition to the word/POS  $n$ -gram models, we also implemented a baseline system to randomly guess the positions of code-switched words in the sentences, and the top  $N$ , herein, means that the system can randomly propose  $N$  candidate positions. Table 3 shows the results for the identification of code-switched words.

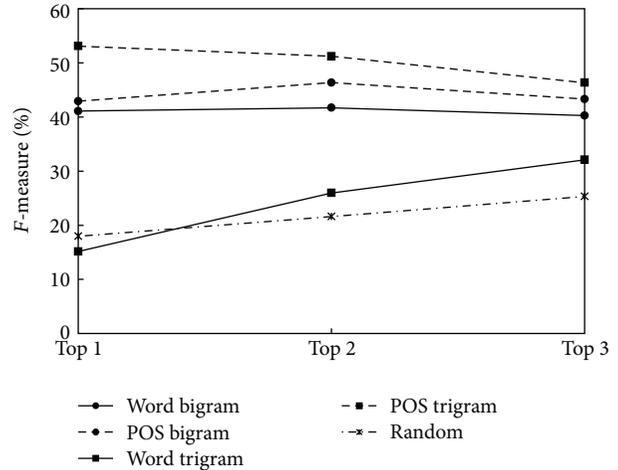
TABLE 4: Results of code-switched word identification in Mandarin-English code-switched sentences.

Methods	Recall	Precision	$F$ -measure
CKIP (FW)	94.70%	95.33%	95.02%
Random (top 1)	17.88%	20.61%	19.15%
Random (top 2)	37.09%	21.37%	27.12%
Random (top 3)	48.34%	18.58%	26.84%

The results show that the  $F$ -measure of the baseline system (Random) was only around 18~25%, indicating that identifying code-switched words is more difficult than identifying code-switched sentences. In addition, the proposed word/POS  $n$ -gram models significantly outperformed Random. For the word-based  $n$ -gram models, the word bigram model achieved an  $F$ -measure of around 41%, which was much better than that of both the word trigram model and Random. Once the POS tags were used to build the language models, both the POS bigram and trigram models outperformed their corresponding word-based models in terms of  $F$ -measure, as well as for recall and precision. This finding indicates that training with the POS tags can reduce the impact of the data sparseness problem. In addition, as shown in Figure 2, the accuracy improvement derived from the trigram model was significantly greater than that from the bigram model, because the trigram model tends to suffer from a more serious data sparseness problem than the bigram model when training data is insufficient. Overall, the best performance of the POS  $n$ -gram models was achieved at an  $F$ -measure of 53.08% (POS trigram, top 1).

Code-switched word identification can also be evaluated by allowing the methods to propose more than one candidate, that is, top 1 to top 3. Table 3 shows that, with more candidates included for verification, more code-switched words were correctly identified, thus dramatically increasing the recall of all methods, but at the cost of reduced precision. Overall, the  $F$ -measure of top 2 was increased for all methods except for the POS trigram, but for top 3, increasing the number of candidates only increased the  $F$ -measure of Random and word trigram.

**4.2.2. Evaluation on Mandarin-English Code-Switched Sentences.** To identify code-switched words in Mandarin-English code-switched sentences, the words associated with the POS tag “FW” (representing a foreign word) by the CKIP word segmentation system were proposed as the answers. The Random system was also implemented to guess the English words in the test sentences. Table 4 shows the comparative results. As expected, the CKIP word segmentation system can provide very precise information for identifying English words in sentences, thus yielding very good performance. Actually, the CKIP system has been under development for over ten years and is still updated periodically. For the Random system, the  $F$ -measure was around 19~27% which was similar to that (18~25%, Table 3) for code-switched word identification in Mandarin-Taiwanese code-switched sentences.

FIGURE 2: Comparative results of top  $N$  performance on code-switched word identification in Mandarin-Taiwanese code-switched sentences.

## 5. Conclusions

This work presents a language modeling method for identifying sentences featuring code-switching and for identifying the code-switched words within those sentences. Experimental results show that the language modeling approach achieved an  $F$ -measure of 80.43% and an accuracy of 79.01% for the detection of Mandarin-Taiwanese code-switched sentences. For the identification of code-switched words in Mandarin-Taiwanese code-switched sentences, the POS  $n$ -gram models outperformed the word  $n$ -gram models, mainly because of the reduced impact of the data sparseness problem. The highest  $F$ -measures (top 1) for the word-based and POS-based models were 41.09% and 53.08%, respectively. For code-switched word identification in Mandarin-English code-switched sentences, the CKIP word segmentation system achieved very high performance (95.02%  $F$ -measure).

Future work will focus on improving system performance by incorporating other effective machine learning algorithms and features, such as sentence structure analysis. The proposed method could also be integrated into practical applications such as a multilingual dialog system to improve effectiveness in dealing with the code-switching problem.

## Acknowledgments

This research was partially supported by the National Science Council (NSC), Taiwan, under Grant nos. NSC 99-2221-E-155-036-MY3, NSC 100-2511-S-003-053-MY2, and NSC 102-2221-E-155-029-MY3, and the “Aim for the Top University Project” of National Taiwan Normal University (NTNU), sponsored by the Ministry of Education, Taiwan. The authors are also grateful to the support of the International Research-Intensive Center of Excellence Program of NTNU and NSC, Taiwan, under Grant no. NSC 102-2911-I-003-301.

## References

- [1] P. Fung and T. Schultz, "Multilingual spoken language processing," *IEEE Signal Processing Magazine*, vol. 25, no. 3, pp. 89–97, 2008.
- [2] D. Hymes, "The ethnography in speaking," in *Anthropology and Man Behaviour*, T. Gladwin and W. C. Sturtevant, Eds., Anthropology Society of Washington, Washington, DC, USA, 1962.
- [3] C. Hoffmann, *An Introduction to Bilingualism*, Longman, London, UK, 1991.
- [4] C. Myers-Scotton, *Social Motivations for Code Switching: Evidence from Africa*, Oxford University Press, New York, NY, USA, 1993.
- [5] M. O. Ayeomoni, "Code-switching and code-mixing: style of language use in childhood in yoruba speech community," *Nordic Journal of African Studies*, vol. 15, no. 1, pp. 90–99, 2006.
- [6] Y. Liu, "Evaluation of the matrix language hypothesis: evidence from Chinese-English code-switching phenomena in blogs," *Journal of Chinese Language and Computing*, vol. 18, no. 2, pp. 75–92, 2008.
- [7] I. Ipsic, N. Pavesic, F. Mihelic, and E. Noth, "Multilingual spoken dialog system," in *Proceedings of the IEEE International Symposium on Industrial Electronics (ISIE '99)*, pp. 183–187, Bled, Slovenia, July 1999.
- [8] H. Holzapfel, "Building multilingual spoken dialogue systems," *Archives of Control Sciences*, vol. 15, no. 4, pp. 555–566, 2005.
- [9] D. C. Lyu, C. N. Hsu, Y. C. Chiang, and R. Y. Lyu, "Acoustic model optimization for multilingual speech recognition," *International Journal of Computational Linguistics and Chinese Language Processing*, vol. 13, no. 3, pp. 363–386, 2008.
- [10] C.-H. Wu, Y.-H. Chiu, C.-J. Shia, and C.-Y. Lin, "Automatic segmentation and identification of mixed-language speech using delta-BIC and LSA-based GMMs," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 14, no. 1, pp. 266–275, 2006.
- [11] J. Y. C. Chan, P. C. Ching, T. Lee, and H. Cao, "Automatic speech recognition of Cantonese-English code-mixing utterances," in *Proceedings of the INTERSPEECH and 9th International Conference on Spoken Language Processing (INTERSPEECH—ICSLP '06)*, pp. 113–116, Pittsburgh, Pa, USA, September 2006.
- [12] W. T. Hong, H. C. Chen, I. B. Liao, and W. J. Wang, "Mandarin/English mixed-lingual speech recognition system on resource-constrained platforms," in *Proceedings of the 21st Conference on Computational Linguistics and Speech Processing (ROCLING '09)*, pp. 237–250, Taichung, Taiwan, September 2009.
- [13] B. H. A. Ahmed and T. P. Tan, "Automatic speech recognition of code switching speech using 1-best rescoring," in *Proceedings of the International Conference on Asian Language Processing (IALP '12)*, pp. 137–140, Hanoi, Vietnam, November 2012.
- [14] N. T. Vu, D. C. Lyu, J. Weiner et al., "A first speech recognition system for Mandarin-English code-switch conversational speech," in *Proceedings of the 37th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '12)*, pp. 4889–4892, Kyoto, Japan, March 2012.
- [15] D.-C. Lyu, C.-L. Zhu, R.-Y. Lyu, and M.-T. Ko, "Language identification in code-switching speech using word-based lexical model," in *Proceedings of the 7th International Symposium on Chinese Spoken Language Processing (ISCSLP '10)*, pp. 460–464, Tainan, Taiwan, December 2010.
- [16] Y.-L. Yeong and T.-P. Tan, "Applying grapheme, word, and syllable information for language identification in code switching sentences," in *Proceedings of the International Conference on Asian Language Processing (IALP '11)*, pp. 111–114, Penang, Malaysia, November 2011.
- [17] Y. Qian, H. Liang, and F. Soong, "A cross-language state sharing and mapping approach to bilingual (Mandarin-English) TTS," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 6, pp. 1231–1239, 2009.
- [18] H.-P. Shen, C.-H. Wu, Y.-T. Yang, and C.-S. Hsu, "CECOS: a Chinese-English code-switching speech database," in *Proceedings of the International Conference on Speech Database and Assessments (Oriental COCODA '11)*, pp. 120–123, Hsinchu, Taiwan, October 2011.
- [19] Y. Seki, N. Kando, and M. Aono, "Multilingual opinion holder identification using author and authority viewpoints," *Information Processing and Management*, vol. 45, no. 2, pp. 189–199, 2009.
- [20] H.-C. Yang, H.-W. Hsiao, and C.-H. Lee, "Multilingual document mining and navigation using self-organizing maps," *Information Processing and Management*, vol. 47, no. 5, pp. 647–666, 2011.
- [21] Y. Zhang, F. S. Tsai, and A. T. Kwee, "Multilingual sentence categorization and novelty mining," *Information Processing and Management*, vol. 47, no. 5, pp. 667–675, 2011.
- [22] C. de Pablo-Sánchez, I. Segura-Bedmar, P. Martínez, and A. Iglesias-Maqueda, "Lightly supervised acquisition of named entities and linguistic patterns for multilingual text mining," *Knowledge and Information Systems*, vol. 35, no. 1, pp. 87–109, 2013.
- [23] F. C. Gey, N. Kando, and C. Peters, "Cross-language information retrieval: the way ahead," *Information Processing and Management*, vol. 41, no. 3, pp. 415–431, 2005.
- [24] M.-F. Tsai, H.-H. Chen, and Y.-T. Wang, "Learning a merge model for multilingual information retrieval," *Information Processing and Management*, vol. 47, no. 5, pp. 635–646, 2011.
- [25] J. J. Jung, "Cross-lingual query expansion in multilingual folksonomies: a case study on flickr," *Knowledge-Based Systems*, vol. 42, pp. 60–67, 2013.
- [26] A. Segev and A. Gal, "Enhancing portability with multilingual ontology-based knowledge management," *Decision Support Systems*, vol. 45, no. 3, pp. 567–584, 2008.
- [27] Y. L. Wu, C. W. Hsieh, W. H. Lin, C. Y. Liu, and L. C. Yu, "Unknown word extraction from multilingual code-switching sentences," in *Proceedings of the 23rd Conference on Computational Linguistics and Speech Processing (ROCLING '11)*, pp. 349–360, Taipei, Taiwan, September 2011.
- [28] C.-H. Wu, C.-H. Liu, M. Harris, and L.-C. Yu, "Sentence correction incorporating relative position and parse template language models," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 18, no. 6, pp. 1170–1181, 2010.
- [29] L. C. Yu, W. C. He, and W. N. Chien, "A language modeling approach to identifying code-switched sentences and words," in *Proceedings of the 2nd CIPS-SIGHAN Joint Conference on Chinese Language Processing (CLP '12)*, pp. 3–8, Tianjin, China, December 2012.
- [30] L.-C. Yu, C.-H. Wu, R.-Y. Chang, C.-H. Liu, and E. Hovy, "Annotation and verification of sense pools in OntoNotes," *Information Processing and Management*, vol. 46, no. 4, pp. 436–447, 2010.

- [31] A. Islam and D. Inkpen, "Near-synonym choice using a 5-gram language model," *Research in Computing Science*, vol. 46, pp. 41–52, 2010.
- [32] L. C. Yu and W. N. Chien, "Independent component analysis for near-synonym choice," *Decision Support Systems*, vol. 55, no. 1, pp. 146–155, 2013.
- [33] W. Y. Ma and K. J. Chen, "Introduction to CKIP Chinese word segmentation system for the first international Chinese word segmentation bakeoff," in *Proceedings of the 2nd ACL SIGHAN Workshop on Chinese Language Processing (CLP '03)*, pp. 168–171, Sapporo, Japan, July 2003.
- [34] W. Y. Ma and K. J. Chen, "A Bottom-up merging algorithm for Chinese unknown word extraction," in *Proceedings of the 2nd ACL SIGHAN Workshop on Chinese Language Processing (CLP '03)*, pp. 31–38, Sapporo, Japan, July 2003.
- [35] A. Stolcke, "SRILM—an extensible language modeling toolkit," in *Proceedings of the 7th International Conference on Spoken Language Processing (ICSLP '02)*, pp. 901–904, Denver, Colo, USA, September 2002.

## Research Article

# Extraction of Belief Knowledge from a Relational Database for Quantitative Bayesian Network Inference

**LiMin Wang**

*Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, Jilin University, ChangChun City, JiLin Province 130012, China*

Correspondence should be addressed to LiMin Wang; wanglim@jlu.edu.cn

Received 22 May 2013; Revised 5 August 2013; Accepted 19 August 2013

Academic Editor: Praveen Agarwal

Copyright © 2013 LiMin Wang. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The problem of extracting knowledge from a relational database for probabilistic reasoning is still unsolved. On the basis of a three-phase learning framework, we propose the integration of a Bayesian network (BN) with the functional dependency (FD) discovery technique. Association rule analysis is employed to discover FDs and expert knowledge encoded within a BN; that is, key relationships between attributes are emphasized. Moreover, the BN can be updated by using an expert-driven annotation process wherein redundant nodes and edges are removed. Experimental results show the effectiveness and efficiency of the proposed approach.

## 1. Introduction

The challenge of extracting knowledge for an uncertain inference is related to the study of Bayesian networks (BNs), which represent one of the key research areas of knowledge discovery and machine learning [1, 2]. The advantages of using prior knowledge by Bayesian statistical methods and clear semantic expression have increased the application of BN in medical diagnosis, manufacturing, and pattern recognition.

A BN consists of two parts: a qualitative part and a quantitative part. The qualitative part denotes the graphical structure of the network, whereas the quantitative part consists of the conditional probability tables (CPTs) in the network. Although BNs are considered efficient inference algorithms, the quantitative part is considered a complex component, and learning an optimal BN structure from existing data has been proven to be a NP-hard problem [3, 4]. Researchers have used qualitative research to improve the efficiency of probabilistic inferences by introducing domain knowledge. Domain experts have proposed ideas on qualitative relations that should be required in the model, such as the requirement of an arc between two nodes. However, domain knowledge may have a negative effect when attributes have similar properties and crossed functional zones. Thus, some attributes are

necessary but redundant. Although these attributes are not biased to the network structure because they do not contribute any negative information, these attributes increase the computational complexity of the network. Given the limited instances for parameter estimation, these necessary attributes should be removed to build a robust structure. Furthermore, the definition and application of expert knowledge in the learning procedure of BN are still unsolved.

In the real world, the widespread use of databases has created a considerable need for knowledge discovery methodologies. Database systems are designed to manage large quantities of information, define structures for storage, and provide mechanisms for mass data manipulation. Functional dependencies (FDs) are a key concept in relational theory and are the foundation of data organization in relational database design. A FD is treated as a constraint that needs to be enforced by the database system to preserve data integrity. A FD can also be used as domain knowledge for knowledge representation and reasoning.

Researchers have recently suggested the linking of the relational database and probabilistic reasoning model to construct a BN from a new perspective. FDs are important data dependencies that provide conditional independency information in relational databases. FDs can be generated

from an entity-relationship diagram instead of being mined from data. Therefore, constructing a BN from FDs is interesting and useful, particularly when data are incomplete and inaccurate. Researchers have indicated the similarities between the BN and relational model. Jaehui and Sang-goo [5] propose a probabilistic ranking model to exploit statistical relationships that exist in relational data of categorical attributes. To quantify the information, the extent of the dependency between correlative attribute values is computed on a Bayesian network. Thimm and Kern-Isberner [6] take an approach of lifting maximum entropy methods to the relational case by employing a relational version of probabilistic conditional logic. They address the problems of ambiguity that are raised by the difference between subjective and statistical views, and develop a comprehensive list of desirable properties for inductive model-based probabilistic inference in relational frameworks. Liu et al. [7] discussed the relationship between fuzzy FD and BN.

Considering the 0/1 data analysis, the association rule mining technique is quite popular and has been studied extensively from computational and objective perspectives by using measures such as frequency and confidence. Many algorithms have been designed to compute frequent and valid association rules. The algorithm proposed in this paper, namely, the tree-augmented Naive Bayes (TAN) classifier with FDs (i.e., TAN-FDA), applies extracted FDs based on association rule analysis. TAN-FDA has two focuses: (1) the use of association rules to infer belief FDs; redundant attributes deduced from FDs can be proven from the viewpoints of information theory and probability theory; (2) the effective simplification and estimation of BN structures and probability distributions, respectively. The feasibility and accuracy of the proposed method are also proven to explain the necessity of finding and eliminating redundant attributes.

This paper is organized as follows. Section 2 introduces related background theories wherein the FD rules of probability are proposed to link FD and probability distribution. Sections 3 and 4 introduce the theoretical foundation used in this paper and the learning procedure of TAN-FDA, respectively. Section 5 compares TAN-FDA with other algorithms. Section 6 concludes.

## 2. Related Background Theory

**2.1. Functional Dependency Rules of Probability.** In the following discussion, we use Greek letters ( $\alpha, \beta, \gamma, \dots$ ) to denote attribute sets. Lower-case letters represent the specific values used by corresponding attributes (e.g.,  $x_i$  represents  $X_i = x_i$ ).  $P(\cdot)$  denotes the probability. Given a relation  $R$  (in a relational database), attribute  $Y$  of  $R$  is functionally dependent on attribute  $X$  of  $R$ , and  $X$  of  $R$  functionally determines  $Y$  of  $R$  (in symbols  $X \rightarrow Y$ ). Armstrong [8] proposed a set of axioms (inference rules) to infer all FDs on a relational database that represents the expert knowledge of organizational data and their inherent relationships. The axioms mainly include the following rules.

- (i) Augmentation rule: if  $\alpha \rightarrow \beta$  is true and  $\gamma$  is a set of attributes, then  $\alpha\gamma \rightarrow \beta\gamma$ .

- (ii) Transitivity rule: if  $\alpha \rightarrow \beta$  and  $\beta \rightarrow \gamma$  are true, then  $\alpha \rightarrow \gamma$ .
- (iii) Union rule: if  $\alpha \rightarrow \beta$  and  $\alpha \rightarrow \gamma$  are true, then  $\alpha \rightarrow \beta\gamma$ .
- (iv) Decomposition rule: if  $\alpha \rightarrow \beta\gamma$  is true, then  $\alpha \rightarrow \beta$  and  $\alpha \rightarrow \gamma$ .
- (v) Pseudotransitivity rule: if  $\alpha \rightarrow \beta$  and  $\gamma\beta \rightarrow \delta$  are true, then  $\alpha\gamma \rightarrow \delta$ .

On the basis of the aforementioned rules, we use the FD rules of probability in [9, 10] to link FD and probability theory. The following rules are included in the FD-probability theory link.

- (i) Representation equivalence of probability: assume that data set  $S$  consists of two attribute sets  $\{\alpha, \beta\}$  and that  $\beta$  can be inferred by  $\alpha$ ; that is, FD  $\alpha \rightarrow \beta$  is true; then the following joint probability distribution is true:

$$P(\alpha) = P(\alpha, \beta). \quad (1)$$

- (ii) Augmentation rule of probability: if  $\alpha \rightarrow \beta$  is true and  $\gamma$  is an attribute set, then the following joint probability distribution is true:

$$P(\alpha, \gamma) = P(\alpha, \beta, \gamma). \quad (2)$$

- (iii) Transitivity rule of probability: if  $\alpha \rightarrow \beta$  and  $\beta \rightarrow \gamma$  are true, then the following joint probability distribution is true:

$$P(\alpha) = P(\alpha, \gamma). \quad (3)$$

- (iv) Pseudotransitivity rule of probability: if  $\gamma\beta \rightarrow \delta$  and  $\alpha \rightarrow \beta$  are true, then the following joint probability distribution is true:

$$P(\alpha, \gamma) = P(\alpha, \gamma, \delta). \quad (4)$$

**2.2. Unrestrictive BN of Naive Bayes and TAN.** BN is a directed acyclic graph that represents a joint probabilistic distribution wherein the nodes denote the domain variables  $U = (X_1, X_2, \dots, X_n)$  and a graph of conditional dependencies denotes a network structure among variables in  $U$ . A conditional dependency connects a child variable with a set of parent variables. This dependency is represented by a table of conditional distributions of the child variable for each combination of the parent variables. Dependency-analysis-based algorithms construct a BN by using dependency information. The mutual information is used to determine the BN structure, which explains the causal relationship between random variables. Hence, dependency-analysis-based algorithms construct a BN by testing the validity of any independence assertions, which lead to a NP-hard computational problem.

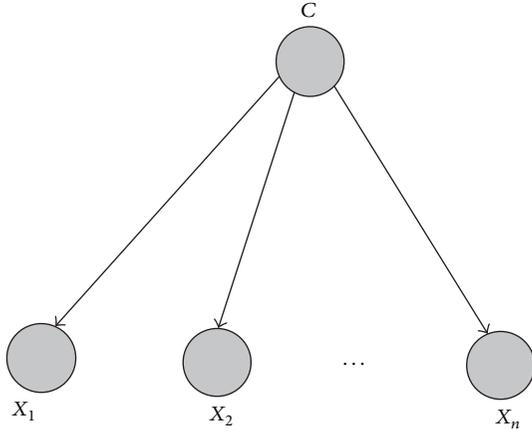


FIGURE 1: Network structure of NB.

**Definition 1.** Entropy is a measure of uncertainty of random variable  $X$ :

$$H(X) = - \sum_{x \in X} P(x) \log P(x), \quad (5)$$

where  $P(x)$  is the probability distribution of  $X$ .

**Definition 2.** Mutual information is the reduction of entropy about variable  $X$  after observing

$$\begin{aligned} I(X; Y) &= H(X) - H(X | Y) \\ &= \sum_{y \in Y} \sum_{x \in X} P(x, y) \log P(x | y) - \sum_{x \in X} P(x) \log P(x). \end{aligned} \quad (6)$$

The mutual information between  $X$  and  $Y$  measures the expected information on  $X$  after observing the value of  $Y$ . If two nodes are dependent in BNs, knowledge of the value of one node will provide information about the value of the other node. Hence, the mutual information between two nodes can show the dependency of the two nodes and the degree of their relationship. To solve the NP-hard computational complexity, Naïve Bayes (NB) [11] assumes that the attributes are independent in a given class. NB has simple structures that contain arcs from the class node to each other node and do not have arcs between other nodes (Figure 1). Although the independence assumption is unrealistic in many practical scenarios, NB has exhibited competitive accuracy with other learning algorithms. Researchers have tried to adjust the Naïve strategy to allow for violations of independence assumptions and improve the prediction accuracy of NB. One straightforward approach in overcoming NB limitations involves the extension of the NB structure to represent explicitly the dependencies among attributes. Friedman et al. [12] presented a compromise representation, that is, TAN, which allows arcs between the children of the class attribute, thereby relaxing the assumption of CI (Figure 2). Given the independence assumption, NB and TAN are both considered restrictive BN.

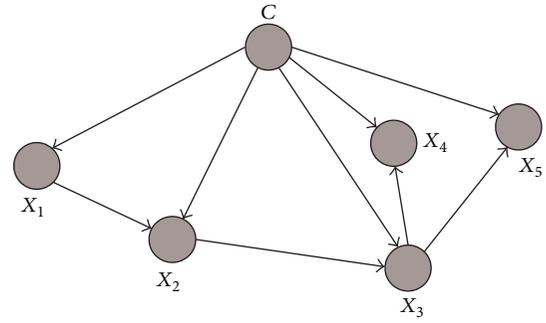


FIGURE 2: Network structure of TAN.

**2.3. Association Rule to Belief Functional Dependency.** Discerning FDs from existing databases is an important issue and has been investigated for a number of years [13]. This issue has recently been addressed in a novel and efficient manner by a data-mining viewpoint. Rather than exhibiting the set of all FDs in a relation, related works have aimed to discover a small cover equivalent of this set. This problem is known as the FD inference problem. Association rules are used to discover the relationships and potential associations of items or attributes in large quantities of data. These rules can be effective in uncovering unknown relationships and providing results that can be the basis of forecasts and decisions. Association rules have also been proven useful tools for enterprises in improving competitiveness and profitability.

Agrawal and Srikant [14] proposed the apriori association rule algorithm, which can discover meaningful item sets and construct association rules within large databases. However, this algorithm generates a large number of candidate item sets from single item sets and requires comparisons to be performed against the whole database level by level in creating association rules. Given a data set  $S$ , an association rule is a rule in the form  $\alpha \rightarrow [s, c]$ , which satisfies the following:

- (i)  $\alpha \subseteq X, \beta \subseteq X$ ,
- (ii)  $\alpha \cap \beta = \Phi$ ,
- (iii)  $s = N_{\alpha\beta}/N_r$  and  $c = N_{\alpha\beta}/N_\alpha$ .

$\alpha$  and  $\beta$  are attribute-value vectors,  $N_r$  is the number of samples in  $S$ ,  $N_{\alpha\beta}$  is the number of samples that contain the set of items  $\alpha \cup \beta$ , and  $N_\alpha$  is the number of samples that contain  $\alpha$ .  $s$  and  $c$  are the support and confidence of the rule, respectively. The pseudocode of the apriori association rule algorithm is shown in Algorithm 1.

Association rules can then be extracted from frequent item sets. When an association rule has a nonzero support and a confidence of 100%, such an association rule can be interpreted as a FD and introduced in this paper for preprocessing. Considering that probability theory is one of the bases of BN, BN requires mass data to ensure precise parameter estimations. For restrictive BN, the structure can only be learned from the training data because the testing data is incomplete, thus making the confidence level relatively low. Some researchers have tried to apply a semisupervised learning or active learning to use useful information in

```

 $C_k$ : candidate item set of size  $k$ 
 $L_k$ : frequent item set of size  $k$ 
 $L_1$  = frequent items;
For ( $k = 1; k \neq \phi; k++$ ) do begin
   $C_{k+1}$  = candidates generated from  $L_k$ ;
  for each transaction  $t$  in database do
    increment the count of all candidates in  $C_{k+1}$  that are contained in  $t$ 
   $L_{k+1}$  = candidates in  $C_{k+1}$  with min support
end
return  $\cup_k L_k$ ;

```

ALGORITHM 1

the testing data and increase the robustness of the final model. These algorithms may cause noise propagation. However, FDs transformed from the association rule will naturally decrease noise to a minimum. Given data set  $U$  with variables  $\{X_1, X_2, \dots, X_n\}$  and class label  $C$ , suppose an association rule  $\alpha \rightarrow \beta[s, c]$  has been deduced from a whole data set and  $\alpha, \beta \subset U$ , such an association rule is transformed to be a FD as  $\alpha \rightarrow \beta$ . From the FD rule of probability, we can obtain the following:

$$P(\alpha) = P(\alpha, \beta). \quad (7)$$

By applying the augmentation rule of probability,

$$P(\alpha, c) = P(\alpha, \beta, c) \quad \text{or} \quad P(\alpha | c) = P(\alpha, \beta | c). \quad (8)$$

Thus, FD  $\alpha \rightarrow \beta$  still occurs in the training data. Thus, we can use the whole data set to extract FDs with high confidence.

### 3. BN Learning Method

*3.1. Redundant Attributes for Restrictive and Unrestrictive BN.* Given that a BN is a complete model for variables and their relationships, a BN can be used to answer probabilistic queries about variables. For example, the network can be used to find updated knowledge on the state of a subset of variables when other variables (e.g., evidence variables) are observed. This process of computing the posterior distribution of variables with evidence is called probabilistic inference. If FD  $x_1 \rightarrow x_2$  exists, the following will be obtained in the augmentation rule of probability:

$$P(x_2 | x_1, c) = \frac{P(x_2, x_1, c)}{P(x_1, c)} = \frac{P(x_1, c)}{P(x_1, c)} = 1. \quad (9)$$

Thereafter, conditional entropy  $H(X_2 | X_1, C) = 0$  and conditional mutual information  $I(X_2; X_1 | C) = H(X_2 | C)$  reach their maximum values. Thus, a definite arc exists between  $X_2$  and  $X_1$  in the restrictive structure of BN.

Furthermore, from the viewpoint of information theory, the information quantity supplied by  $X_1$  and  $X_2$  to any other attribute sets  $Y$  is expressed as follows:

$$\begin{aligned}
 I(Y; X_1, X_2) &= H(Y) - H(Y | X_1, X_2) \\
 &= \sum_{y \in Y} P(y) \log P(y) - \sum_{y \in Y} P(y, x_1, x_2) \log P(y | x_1, x_2) \\
 &= \sum_{y \in Y} P(y) \log P(y) - \sum_{y \in Y} P(y, x_1) \log P(y | x_1) \\
 &= H(Y) - H(Y | X_1) = I(Y; X_1).
 \end{aligned} \quad (10)$$

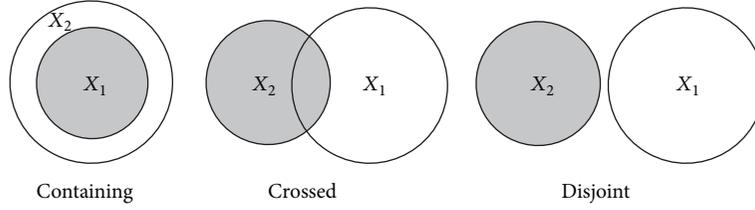
Hence, from the viewpoint of probability and information theory,  $X_2$  is a redundant attribute that does not need to be considered during structure learning.

If the information supplied by  $X_1$  is implicated in the information supplied by  $X_2$ ,  $X_1$  and  $X_2$  have a strong relationship, but  $X_1$  is redundant for modeling. By contrasts, if the information boundaries of  $X_1$  and  $X_2$  are crossed, a relationship exists at some point between  $X_1$  and  $X_2$ . If the boundaries are disjointed,  $X_1$  and  $X_2$  are independent. This result is clearly displayed in Figure 3.

*3.2. Structure Simplification for BN Learning.* In relational database theory, the minimal FD set can be inferred by canonical cover analysis. Let  $F_c$  be a canonical cover for a set of simple FDs over  $U$  and closure of  $\alpha$  denoted as  $\alpha^+$ , which represents all attributes functionally determined by  $\alpha$ . Redundant attributes can be found from the viewpoints of chain formula and FD rules of probability.

A joint probability distribution can be expressed by a chain formula that includes construction information for a BN. However, constructing the structure solely from a joint probability distribution without using any conditional independencies is impractical because such an approach will require an exponentially large number of variable combinations. We can use FDs to simplify the structure of a BN which is represented by a set of probability distributions [15] (see Algorithm 2).

We use the following example to elaborate the algorithm.


 FIGURE 3: Information boundaries of attributes  $X_1$  and  $X_2$ .

```

Input: FDs and probability distribution sets  $(S_1, S_2, \dots, S_n)$ ,
where  $S_1 = P(A_1), S_2 = P(A_2 | A_1), \dots, S_n = P(A_n | A_1 A_2, \dots, A_{n-1})$ .
Output: A chain formula with independence conditions implied by FDs.
Begin
For every  $S_k$  ( $k > 2$ ) Do
  If FD =  $x_k \rightarrow a_k$ , substitute  $P(a_k | a_1 a_2, \dots, a_{k-1})$  with  $P(a_k | x_k)$ ,
  where  $x_k$  is a minimal set of  $(a_1 a_2, \dots, a_{k-1})$ .
End For
End.
    
```

ALGORITHM 2

*Example 3.* Let  $R(U + F)$  be a probabilistic scheme;  $U = \{X_1, X_2, X_3, X_4\}$  and  $F = \{x_2 \rightarrow x_1, x_3 \rightarrow x_4\}$  are sets of FDs over  $R$ .

According to the order of  $(X_2, X_3, X_1, X_4)$ , the joint probability distribution  $P(x_1, x_2, x_3, x_4)$  will be expressed as follows:

$$P(x_1, x_2, x_3, x_4) = P(x_2) P(x_3 | x_2) \times P(x_1 | x_2, x_3) P(x_4 | x_1, x_2, x_3). \quad (11)$$

The following results can be generated based on Algorithm 2.

- (1) By using FD  $x_2 \rightarrow x_1$ ,  $P(x_1 x_2, x_3) = P(x_1 x_2)$ .
- (2) By using FD  $x_3 \rightarrow x_4$ ,  $P(x_4 x_1, x_2, x_3) = P(x_4 x_3)$ .

The joint probability distribution can then be simplified as follows:

$$P(x_1, x_2, x_3, x_4) = P(x_2) P(x_3 | x_2) P(x_1 | x_2) P(x_4 | x_3). \quad (12)$$

The corresponding BN structure of the joint probability distribution is shown in Figure 4(a). From the FD rule of probability, given that  $P(x_1 | x_2) = P(x_4 | x_3) = 1$  is true, (11) is changed into the following expression:

$$P(x_1, x_2, x_3, x_4) = P(x_2) \cdot P(x_3 | x_2) \cdot 1.1 = P(x_2) \cdot P(x_3 | x_2). \quad (13)$$

The corresponding structure is shown in Figure 4(b).

**Theorem 4** (see [9]). Given two attribute sets  $\{\alpha, \beta\}$  and class label  $C$ , if  $\alpha \rightarrow \beta$  is true, then the following conditional probability distribution is true:

$$P(c | \alpha) = P(c | \alpha, \beta). \quad (14)$$

**Theorem 5** (see [16]). Let  $U$  be a set of attributes and  $F$  an acyclic set of simple FDs over  $U$ . One canonical cover will exist for  $F$ ; that is,  $F_1 = F_2$  if  $F_1$  and  $F_2$  are canonical covers for  $F$ .

The chosen canonical cover is the set of minimal dependencies. Such a cover is information lossless and is considerably smaller than the set of all valid dependencies. These qualities are particularly important for the end user because of the provision of relevant knowledge wherein redundancy is minimized and extraneous information is discarded.

**Lemma 6.** Let  $F_c$  be a canonical cover for  $F$ , which is a set of simple FDs, and let  $\alpha$  and  $\beta$  be the union of the left-hand sides of all dependencies in  $F_c$  and  $F$ , respectively. The following conditional probability distribution is true:

$$P(c | \alpha) = P(c | \beta^+). \quad (15)$$

**Lemma 7.** Let  $U$  be a set of attributes and  $C$  the class label. Let  $F_c$  be a canonical cover for  $F$ , which is a set of simple FDs, and let  $\alpha$  be the union of the left-hand sides of all dependencies in  $F_c$ . The following conditional probability distribution is true:

$$P(c | \alpha, U - \alpha^+) = P(c | U), \quad (16)$$

where  $\{U - \alpha^+\}$  represents the difference set between  $U$  and  $\alpha^+$ . Therefore, the difference set  $\{\alpha^+ - \alpha\}$  is redundant for classification. From Lemma 7 and chain formula rule,

$$\begin{aligned} & P(c | x_1, x_2, x_3, x_4) \\ &= \operatorname{argmax} P(x_2 | c) P(x_3 | x_2, c) P(x_1 | x_2, x_3, c) \\ &\quad \times P(x_4 | x_1, x_2, x_3, c) P(c) \\ &= \operatorname{argmax} P(x_2 | c) P(x_3 | x_2, c) P(x_1 | x_2) \\ &\quad \times P(x_4 | x_3) P(c). \end{aligned} \quad (17)$$

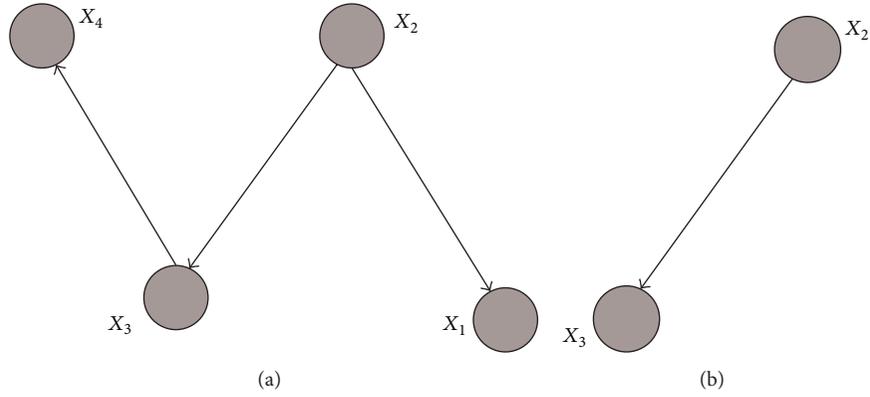


FIGURE 4: Two simplification steps of BN structure by applying FDs.

The corresponding structure is shown in Figure 5(a). By applying the FD rule of probability, (17) is changed into the following:

$$\begin{aligned}
 & P(c, x_1, x_2, x_3, x_4) \\
 &= \operatorname{argmax} P(x_2 | c) P(x_3 | x_2, c) \cdot 1.1 \cdot P(c) \quad (18) \\
 &= \operatorname{argmax} P(x_2 | c) P(x_3 | x_2, c) P(c).
 \end{aligned}$$

The corresponding structure is shown in Figure 5(b).

#### 4. TAN-FDA

TAN allows tree-like structures to be used in representing dependencies among attributes. The class node directly points to all attribute nodes, and an attribute node can have only one parent from another attribute node (in addition to the class node).

The architecture of the TAN model can be depicted as shown in Algorithm 3.

The proposed TAN-FDA has three phases: drafting, thinning, and thickening. Given that TAN performs well in the experimental study, the drafting phase does not start from the empty graph but from the TAN structure that is expected to be close to the correct graph. Moreover, we propose another method for performing the thinning phase. This new methodology uses a data-mining technique to infer the number of FDs, which describe the correlation of events and can be viewed as probabilistic rules. Two events are correlated if they are frequently observed together. Each FD can be used to remove redundant attributes, thus simplifying the network structure. Some relationships can be neglected for TAN with a limited number of training samples. However, after the thinning phase, the attribute space decreases, and the training samples can provide useful information. In the thickening phase, the conditional mutual information is computed to obtain a new TAN structure with the rest of the attributes. We then compared the differences between TAN structures before and after the thickening phase. If a new edge exists, the edge is added to the graph in the thickening phase. The graph produced by this phase will contain all the edges of the underlying dependency model.

TABLE 1: Description of the data sets used in the experiments.

Data set	Size	Attributes	Number of classes
Anneal	798	39	6
Balance-scale	625	5	3
German	1000	24	2
Glass	214	10	7
Heart-c	303	14	5
Ionosphere	351	34	2
Pima Indians	768	8	2
Audiology	226	70	24
Sonar	208	61	2

The learning procedure of the TAN-FDA algorithm is described as shown in Algorithm 4.

#### 5. Experiments

To verify the efficiency and effectiveness of the proposed TAN-FDA, we conduct experiments on nine data sets from the UCI machine-learning repository (Table 1). For each benchmark data set, we compare two Bayes models with the selected attributes. The following abbreviations are used for the different classification approaches: SNB-FSS [17]: selective Naive Bayes classifier with forward sequential selection; TAN-CFS [18]: tree-augmented Naive Bayes classifier with classical floating search.

SNB-FSS selects a subset of attributes by using leave-one-out cross validation as a selection criterion and establishes an NB with these attributes. SNB-FSS uses the reverse search direction; that is, SNB-FSS iteratively adds the attribute to improve accuracy, starting with the empty set of attributes. Independence is assumed among the resulting attributes in a given class.

TAN-CFS includes new features that maximize the criterion  $J$  by means of the sequential forward selection procedure and starts from the current feature set. Thereafter, the conditional exclusions of previously updated subsets are implemented. If no feature can be excluded, the algorithm proceeds with the sequential forward selection algorithm.

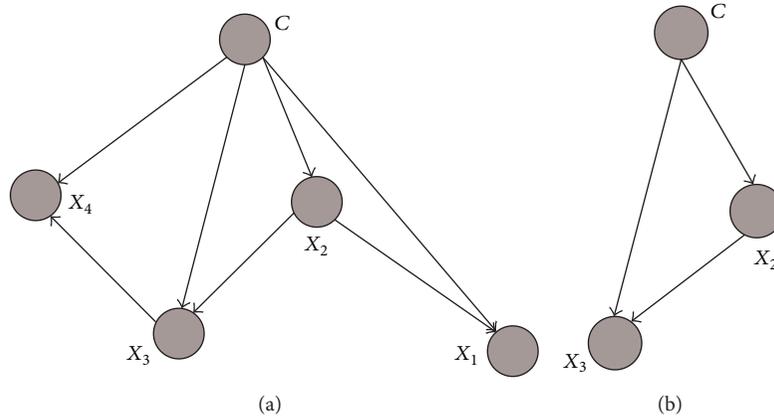


FIGURE 5: Two simplification steps of restrictive BN by applying FDs.

Input: Training set  $T$ , attribute set  $X = \{X_1, \dots, X_n\}$ , and class  $C$ .  
Output: TAN constructed by conditional mutual information metric.  
Step 1. Compute the conditional mutual information  $I(X_i; X_j | C)$  between each pair of attributes  $\{X_i, X_j\}, i \neq j$ .  
Step 2. Build a complete undirected graph wherein the vertices are attributes  $\{X_1, \dots, X_n\}$ . Annotate the weight of an edge connecting  $X_i$  to  $X_j$  by  $I(X_i; X_j | C)$ .  
Step 3. Build a maximum weighted spanning tree.  
Step 4. Transform the resulting undirected tree to a directed tree by choosing a root node and setting the direction of all edges outward from the directed tree.  
Step 5. Build a TAN model by adding a vertex labeled by  $C$  and adding an arc from  $C$  to each  $X_i$ .

## ALGORITHM 3

The floating methods are allowed to correct the wrong decisions made in the previous steps. These methods also approximate the optimal solution better than sequential feature selection methods.

The current implementation of TAN-FDA is limited to categorical data. Hence, we assess only the relative capacities of these algorithms with respect to categorical data, and all numeric attributes are discretized. When MDL discretization [19], which is a common discretization method for NB, is used to discretize quantitative attributes within each cross-validation fold, many attributes will have only one value. In these experiments, we discretize quantitative attributes by using three-bin equal-frequency discretization prior to classification.

The base probabilities are estimated by using  $m$ -estimation [20] ( $m = 1$ ), which often results in more accurate probabilities than the Laplace estimation for NB and TAN. The above experiments are coded in Matlab 7.0 on an Intel Pentium 2.93 GHz computer with 1 GB of RAM.

The main advantage of the proposed method is that it uses FD to simplify the learning procedure and builds a robust BN structure. For different testing samples, different FDs may be applied and the final BN structure may differ greatly, which make TAN-FDA much more flexible. During our experiments, we try to use the most general FDs as inductive rules wherein the number of attributes (denoted

as  $l$  in the following discussion) of the left side of each FD is less than three. We first study the performance of the state-of-the-art classifiers NB and TAN to reveal how performance varies with changing  $l$ . To explore how the classification performance of TAN-FDA compares with SNB-FSS and TAN-CFS, we estimate  $P_{\text{cost}}$ , which is the mean posterior probability of the submodels. The probabilistic costing is equivalent to the accumulated code length of the total test data encoded by an inferred model. Given that an optimal code length can only be achieved by encoding the real probability distribution of the test data, a smaller probabilistic costing will indicate a better performance on the overall probabilistic prediction than a larger probabilistic costing. For each benchmark data set, the classification performance is evaluated by fourfold and tenfold cross-validation.

**5.1. Comparison with State-of-the-Art Algorithms.** We choose NB and TAN as comparator algorithms because they are relatively unparameterized and can readily produce clearly understood performance outcomes. We first consider the relative performance when the induction rules or FDs take different  $l$  values, that is, one or two. In an attempt to assess these predictions, we calculated the mean of error, bias, and variance for NB and TAN over the nine data sets. The experimental results are presented in Tables 2 and 3. We observe

Input: Training set  $T_1$ , testing set  $T_2$ , and attribute set  $X = \{X_1, \dots, X_n\}$ .  
Output: Restrictive TAN model.  
Step 1. In the drafting phase, the TAN model is used as the basic structure.  
Step 2. Mine association rules from data set  $\{T_1 + T_2\}$ , and transform these rules into FDs. Thereafter, obtain the closure of FDs.  
Step 3. In the thinning phase, remove redundant attributes and corresponding edges that originate from these attributes. Thereafter, obtain the simplified structure.  
Step 4. Learn the TAN model as the mapping structure with the rest of the attributes.  
Step 5. In the thickening phase, compare the mapping structure and simplified structure. If a new edge exists in the mapping structure, add the edge to the simplified structure.

ALGORITHM 4

TABLE 2: Comparative error, bias, and variance for NB.

Classifier	NB ( $l = 1$ )	NB ( $l = 2$ )	NB
Mean error	0.109	0.097	0.112
Mean bias	0.121	0.117	0.135
Mean variance	0.035	0.037	0.032

TABLE 3: Comparative error, bias, and variance for TAN.

Classifier	TAN ( $l = 1$ )	TAN ( $l = 2$ )	TAN
Mean error	0.093	0.087	0.105
Mean bias	0.105	0.093	0.124
Mean variance	0.028	0.033	0.027

TABLE 4: Comparative error, bias, and variance for TAN-FDA versus SNB-FSS.

Classifier	TAN-FDA ( $l = 1$ )	TAN-FDA ( $l = 2$ )	SNB-FSS
Mean error	0.091	0.088	0.097
Mean bias	0.157	0.166	0.143
Mean variance	0.022	0.032	0.036

TABLE 5: Comparative error, bias, and variance for TAN-FDA versus TAN-CFS.

Classifier	TAN-FDA ( $l = 1$ )	TAN-FDA ( $l = 2$ )	TAN-CFS
Mean error	0.091	0.088	0.089
Mean bias	0.157	0.166	0.120
Mean variance	0.022	0.032	0.044

that increasing  $l$  from one to two consistently decreases bias at the cost of an increase in variance. This trade-off delivers low errors for NB and TAN. If our reasoning on the expected bias profiles of these algorithms is accepted, the performance of NB or TAN should increase with increasing data quantity, and unrestricted BN classifiers should also achieve low errors.

**5.2. Comparison with SNB-FSS and TAN-CFS.** Tables 4 and 5 show the mean of error, bias, and variance results of TAN-FDA, SNB-FSS, and TAN-CFS. TAN-FDA has higher biases, lower variances, and lower mean errors than SNB-FSS and TAN-CFS. When  $l$  increases from one to two, fewer instances may satisfy the stopping criterion of the

association rule. Moreover, the rules or extracted FDs may only be coincidental and are not real domain knowledge. The occurrence of low variance is less obvious than the occurrence of high mean bias. This phenomenon remains an interesting unexplained topic that is worthy of further investigation.

To obtain  $P_{\text{cost}}$  for BN, the probabilistic prediction for each test instance is calculated by arithmetically averaging the probabilistic predictions submitted by each iteration. We observe that TAN-FDA has achieved better (lower) probabilistic costing than SNB-FSS and TAN-CFS in almost all data sets in terms of logarithm of probability  $P_{\text{cost}}$  bit costing (Figure 6). The superior performance of TAN-FDA on probabilistic prediction can be attributed to the belief that FDs are extracted from a whole data set, irrelevant attributes are not included, and classifiers are based on subsets of selected attributes. This restricted network structure maximizes the classification performance by removing irrelevant attributes and relaxing independence assumptions between correlated attributes. The computational demands for determining the network structure are low, particularly if a large number of attributes are available.

The size of the CPTs of the nodes increases exponentially with the number of parents, thus possibly resulting in an unreliable probability estimate of the nodes that have a large number of parents. However, the introduction of FDs reduces this negative effect significantly.

## 6. Conclusion

For high-dimensional data, only very low-dimensional forms of BN are robust. FDs provide a novel way of solving this common problem that exists in machine-learning techniques. Moreover, we have established that higher-dimensional variants are likely to deliver greater accuracy than lower-dimensional alternatives when provided with reliable domain knowledge. Thus, a promising direction for future research is the development of computationally efficient techniques for approximating TAN-FDA with high  $l$  values.

A further unresolved issue is the manner of selecting an appropriate  $l$  value for any specific data set  $T$ . Furthermore, a number of techniques have been developed for extending TAN to handle numeric data [21]. Extending the current study to the general TAN-FDA framework is needed.

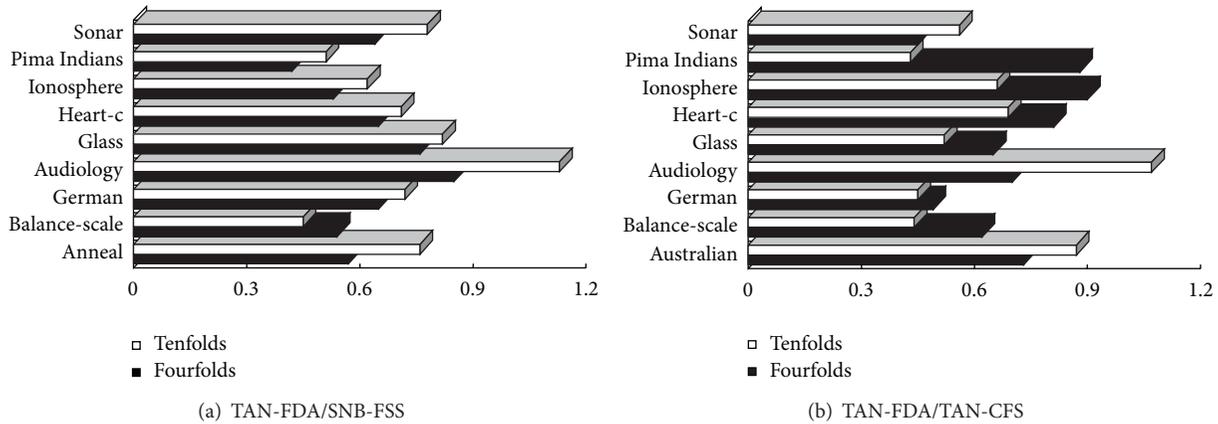


FIGURE 6: Comparison of TAN-FDA to SNB-FSS and TAN-CFS.

We have presented a strategy for deriving FDs and removing redundant nodes. However, mining relationships between different FDs should also be important. Reducing the errors of BNs has been proven possible by appropriate feature selection and submodel weighting [22]. Therefore, exploring efficient methods for removing redundant FD parts and obtaining high  $l$  values is significant. If fast classification is required and training time is not constrained, approaches that use search methods to select small numbers of FDs from a TAN-FDA model are likely to be appropriate. If sufficient training time is available, searching for appropriate  $l$  values will be useful.

For forward sequential selection (FSS) to consider joining two attributes, FSS must first construct a classifier wherein one of the attributes is used independently. However, on exclusive—or, a Bayesian classifier with only one relevant attribute will not perform better than a simple guess on the most frequent class, and correct relevant attributes will rarely be joined. By contrast, classical floating search (CFS) starts with the TAN classifier of all attributes and considers all pairs of joins. CFS always considers joining two relevant attributes (specifically on larger data sets) and has high accuracy. A potential weakness of the algorithm for joining attributes is that to join more than two attributes, two attributes must first be joined before joining other attributes. This process will not occur unless the formation of the first pair results in an increase in accuracy. This problem is common in many algorithms. In this paper, we have developed a generative learning algorithm that generalizes the principles underlying TAN-FDA. We have shown that searching for dependencies among attributes while learning Bayesian classifiers results in significant increases in accuracy. All redundant attributes will be considered only once to improve the accuracy of the Bayesian classifier when more than two interacting attributes are present.

## Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant no. 61272209) and Postdoctoral Science Foundation of China (Grant no. 20100481053).

## References

- [1] V.-F. Joan, S. Juan, and S.-O. Emilio, "Expert system for predicting unstable angina based on Bayesian networks," *Expert Systems With Applications*, vol. 40, no. 12, pp. 5004–5010, 2013.
- [2] J. Yu Zhang and H. Song, "Bayesian-network-based reliability analysis of PLC systems," *IEEE Transactions on Industrial Electronics*, vol. 60, no. 11, pp. 5325–5336, 2013.
- [3] P. Larrañaga, H. Karshenas, C. Bielza, and R. Santana, "A review on evolutionary algorithms in Bayesian network learning and inference tasks," *Information Sciences*, vol. 233, pp. 109–125, 2013.
- [4] S. S. Young and L. A. Sung, "Bayesian network analysis for the dynamic prediction of early stage entrepreneurial activity index," *Expert Systems With Applications*, vol. 40, no. 10, pp. 4003–4009, 2013.
- [5] P. Jaehui and L. Sang-goo, "Pragmatic correlation analysis for probabilistic ranking over relational data," *Expert Systems with Applications*, vol. 40, no. 7, pp. 2649–2658, 2013.
- [6] M. Thimm and G. Kern-Isberner, "On probabilistic inference in relational conditional logics," *Logic Journal of the IGPL*, vol. 20, no. 5, pp. 872–908, 2012.
- [7] W.-Y. Liu, K. Yue, and W.-H. Li, "Constructing the Bayesian network structure from dependencies implied in multiple relational schemas," *Expert Systems with Applications*, vol. 38, no. 6, pp. 7123–7134, 2011.
- [8] W. W. Armstrong, "Dependency structures of data base relationships," in *Proceedings of the International Federation for Information Processing Congress (IFIP '74)*, pp. 580–583, 1974.
- [9] L. Wang, G. Yao, and X. Li, "Extracting logical rules and attribute subset from confidence domain," *Information*, vol. 15, no. 1, pp. 173–180, 2012.
- [10] L. M. Wang, X. H. Wei, and L. Y. Dong, "Bayesian network inference based on functional dependency mining of relational database," *Information*, vol. 15, no. 6, pp. 2441–2446, 2012.
- [11] Z. Xie, W. Hsu, and Z. Liu, "SNNB: a selective neighborhood based naive Bayes for lazy learning," in *Proceedings of the 6th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining*, pp. 104–114, 2002.
- [12] N. Friedman, D. Geiger, and M. Goldszmidt, "Bayesian network classifiers," *Machine Learning*, vol. 29, no. 2-3, pp. 131–163, 1997.

- [13] F. Ferrarotti, S. Hartmann, and S. Link, "Reasoning about functional and full hierarchical dependencies over partial relations," *Information Sciences*, vol. 235, pp. 150–173, 2013.
- [14] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules in large databases," in *Proceedings of the 20th International Conference on Very Large Data Bases*, pp. 487–499, 1994.
- [15] S. S. Liao, H. Q. Wang, Q. D. Li, and W. Y. Liu, "A functional-dependencies-based Bayesian networks learning method and its application in a mobile commerce system," *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 36, no. 3, pp. 660–671, 2006.
- [16] J. Lechtenbörger, "Computing unique canonical covers for simple FDs via transitive reduction," *Information Processing Letters*, vol. 92, no. 4, pp. 169–174, 2004.
- [17] I. Inza, P. Larrañaga, and B. Sierra, "Feature subset selection by Bayesian networks: a comparison with genetic and sequential algorithms," *International Journal of Approximate Reasoning*, vol. 27, no. 2, pp. 143–164, 2001.
- [18] F. Pernkopf and P. O'Leary, "Floating search algorithm for structure learning of Bayesian network classifiers," *Pattern Recognition Letters*, vol. 24, no. 15, pp. 2839–2848, 2003.
- [19] U. M. Fayyad and K. B. Irani, "Multi-interval discretization of continuous valued attributes for classification learning," in *Proceedings of the 13th International Conference on Artificial Intelligence*, pp. 1022–1029, 1993.
- [20] R. Kohavi, "Scaling up the accuracy of naive-Bayes classifiers: a decision-tree hybrid," in *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining*, pp. 202–207, 1996.
- [21] M. J. Flores, J. A. Gámez, A. M. Martínez, and J. M. Puerta, "Handling numeric attributes when comparing Bayesian network classifiers: does the discretization method matter?" *Applied Intelligence*, vol. 34, no. 3, pp. 372–385, 2011.
- [22] P. A. D. Castro and F. J. von Zuben, "Learning Bayesian networks to perform feature selection," in *Proceedings of the International Joint Conference on Neural Networks (IJCNN '09)*, pp. 467–473, June 2009.

## Research Article

# Robust Quadratic Regression and Its Application to Energy-Growth Consumption Problem

Yongzhi Wang,<sup>1</sup> Yuli Zhang,<sup>2</sup> Fuliang Zhang,<sup>3</sup> and Jining Yi<sup>3,4</sup>

<sup>1</sup> College of Instrumentation & Electrical Engineering, Jilin University, Changchun 130061, China

<sup>2</sup> Department of Automation, TNLIST, Tsinghua University, Beijing 100084, China

<sup>3</sup> Development and Research Center of China Geological Survey, Beijing 100037, China

<sup>4</sup> School of Earth Sciences and Resources, China University of Geosciences, Beijing 100083, China

Correspondence should be addressed to Yongzhi Wang; [iamwangyongzhi@126.com](mailto:iamwangyongzhi@126.com)

Received 1 May 2013; Accepted 8 August 2013

Academic Editor: Yudong Zhang

Copyright © 2013 Yongzhi Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We propose a robust quadratic regression model to handle the statistics inaccuracy. Unlike the traditional robust statistic approaches that mainly focus on eliminating the effect of outliers, the proposed model employs the recently developed robust optimization methodology and tries to minimize the worst-case residual errors. First, we give a solvable equivalent semidefinite programming for the robust least square model with ball uncertainty set. Then the result is generalized to robust models under  $l_1$ - and  $l_\infty$ -norm criteria with general ellipsoid uncertainty sets. In addition, we establish a robust regression model for per capital GDP and energy consumption in the energy-growth problem under the conservation hypothesis. Finally, numerical experiments are carried out to verify the effectiveness of the proposed models and demonstrate the effect of the uncertainty perturbation on the robust models.

## 1. Introduction

Traditional regression analysis is a useful tool to model the linear or nonlinear relationship between the observed data. In the simplest linear regression model, there is only one explanatory variable  $x$  (the regressors) and one dependent variable  $y$  (the regressand) that is assumed to be an affine function of  $x$ ; it is further extended to a polynomial regression model where  $y$  is an  $n$ th order polynomial of  $x$ . In this case, the corresponding multivariate regression model contains more than one explanatory variable. To make the regression models work well, there are several specific assumptions on the model and the observed data.

Consider the following standard multivariate linear regression model:

$$y_i = \sum_{j=1}^n x_{ij} + \varepsilon_i, \quad i = 1, \dots, m, \quad (1)$$

where  $(x_i, y_i)_{i=1}^m$  are given observed data and  $\varepsilon$  is a random error vector. Assuming that  $(\varepsilon_i)_{i=1}^m$  have zero mean and constant variance, they are independent of each other.

Besides the assumption on the random errors, there is another important weak exogeneity assumption that the explanatory variables are known deterministic values. Under this assumption, one can arbitrarily transform their values and construct any complex function relationship between the regressors and the regressand. For example, in this case the polynomial regression is merely a linear regression with regressors  $\{x_i^1, x_i^2, \dots, x_i^p\}$ .

Although this weak exogeneity assumption makes the linear regression model very powerful to fit the given data or predict the regressand for given known regressors, it may lead to overfitting or inconsistent estimations [1]. Actually this assumption may be quite unreasonable in some case. For instance, in the process of collecting data, there is often unavoidable observation noise that makes the observed data quite inaccurate. Furthermore, in statistics the incomplete sampling approach sometimes can only give an approximation of the real values.

Researches on regression models with imprecise data have been reported. One way to handle the noisy observation is the measure error model or the errors-in-variable model, where it is assumed that there exist some unknown latent (or

true) variables that follow the true functional relationship, and the actual observations are affected by certain random noise [2]. Based on different assumptions about random noise, there are a variety of regression models, such as the method of moments [3] that is based on the third-(or higher-) order joint cumulants of observable variables and the Deming regression [4] assuming that the ratio of the noise variance is known. A brief historical overview of linear regression with errors in variables can be found in [5].

In addition to the errors-in-variable models, motivated by the robust optimization theory under uncertainty, studies on the robust regression models are reported. In such case, the perturbations are deterministic and unknown but bounded. Ghaoui and Lebret [6] study the robust linear regression with bounded uncertainty sets under least square criterion. They utilize the second-order cone programming (SOCP) and semi-definite programming to minimize the worst-case residual errors. Shivaswamy et al. [7] propose SOCP formulations for robust linear classification and regression models when the first two moments of the uncertain data are computable. Ben-Tal et al. [8] provide an excellent framework of robust linear classification and regression. Based on general assumption on the uncertainty sets, they provide explicit equivalent formulations for robust least squares,  $l_1$ ,  $l_\infty$ , and Huber penalty regressions. For more results regarding robust classification and regression that are similar to this work, we refer the readers to [9–11]. Also according to [8], the traditional robust statistic approaches (see [12]), which try to reject the outliers in data, are different from the point of view in this paper as the authors here intend to minimize the maximal (worst-case) residual errors. However in order to overcome the confliction, a two-step approach can be easily implemented. First, the outliers are identified, and related data is removed. Then, our proposed method is applied in order to safely eliminate the effect generated from imprecise data. We employ this approach in the real energy-growth regression problem in Section 3.

Besides the regression models, there are a wide variety of forecasting models, such as support vector machines, decision tree, neural network, and Bayes classifier. For example, [13] utilizes the support vector machine based on trend-based segmentation method for financial time series forecasting. The proposed models have been tested by using various stocks from America stock market with different trends [14]. [14] proposes a new adaptive local linear prediction method to reduce the parameters uncertainties in the prediction of a chaotic time series. Real hydrological time series are used to validate the effectiveness of the proposed methods. More related literatures can be found in [15] (chaotic time series analysis), [16] (fractal time series), and [17] (knowledge-based Green's kernel for support vector regression). Compared with these models, we focus on the handling of the statistics inaccuracy. The regression model is an appropriate basis to develop effective and tractable robust models.

In this paper, we try to extend the robust linear regression model to general multivariate quadratic regression and provide equivalent tractable formulations. Different from the simple extension from the classical linear model to classical polynomial (even general nonlinear) models under the

weak exogeneity assumption, the perturbation of explanatory variables in the quadratic terms will affect the model in a complex nonlinear manner. Although [8, 12] have discussed the robust polynomial interpolation problem, only an upper bound and the corresponding suboptimal coefficients are given. They further conjecture that the proposed problem cannot be solved exactly in polynomial time. Our proposed robust multivariate quadratic regression model in this paper also needs to solve a complex biquadratic min-max optimization problem. However, under certain assumption on the uncertainty sets, we can obtain a series of equivalent semidefinite programming formulations for robust quadratic regression under different residual error criteria.

In particular, we first extend the traditional quadratic regression model by introducing the separable ball (2-norm) uncertainty set and formulate the optimal robust regression problem as a min-max problem that tries to minimize the maximal residual error. By utilizing the S-lemma [18] and Schur complement lemma, we provide an equivalent semidefinite programming formulation for the robust least square quadratic regression model with ball uncertainty set. This result is then generalized to models with general ellipsoid uncertainty sets and under the  $l_1$ -,  $l_\infty$ -norm criteria. Furthermore the robust quadratic regression models are applied to the economic growth and energy consumption regression problem. We take the per capital GDP as the explanatory variable and the per capital energy consumption as the dependent variable. Under the conservation hypothesis, we establish a corresponding robust model. Finally we test the proposed model on different history data sets and compare our models with the classical regression models.

The paper proceeds as follows. In Section 2, we present a general robust quadratic regression model, give a solvable equivalent semi-definite programming for the robust least square quadratic regression model with ball uncertainty set, and further generalize the result. In Section 3, the proposed models are applied to the energy-growth problem. Numerical experiments are carried out in Section 4, and Section 5 concludes this paper and gives future research directions.

## 2. Robust Quadratic Regression Models

**2.1. General Robust Models.** Consider the standard multivariate quadratic regression model:

$$y = x^T Q x + 2\alpha^T x + \beta, \quad (2)$$

where  $x \in R^n$  denotes the  $n$ -dimension explanatory data,  $y \in R$  denotes the dependent data, and  $Q \in R^{n \times n}$ ,  $\alpha \in R^n$ , and  $\beta \in R$  are unknown coefficients that will be determined based on certain minimal criteria.

Given a set of data  $D = [X; Y^T] \in R^{(n+1) \times m}$ , where  $X = [x_1, \dots, x_m] \in R^{n \times m}$  and  $Y = [y_1; \dots; y_m] \in R^m$ , we utilize the  $p$ -norm to measure the prediction error

$$e_p(\alpha, \beta, Q) = \left\| \begin{pmatrix} y_1 - x_1^T Q x_1 - 2\alpha^T x_1 - \beta \\ \vdots \\ y_m - x_m^T Q x_m - 2\alpha^T x_m - \beta \end{pmatrix} \right\|_p. \quad (3)$$

In traditional regression models, we assume that the explanatory data are precise and reliable. Based on this weak exogeneity assumption, the quadratic regression can be expressed as the following linear regression:

$$\min_{\alpha, \beta, Q} \left\| \begin{pmatrix} y_1 - Q \circ X_1 - 2\alpha^T x_1 - \beta \\ \vdots \\ y_m - Q \circ X_m - 2\alpha^T x_m - \beta \end{pmatrix} \right\|_p, \quad (4)$$

where  $X_i = x_i x_i^T$  are the problem data and the linear operator  $\circ$  for matrix  $A$  and  $B \in R^{s \times l}$  is defined as  $A \circ B = \sum_{i=1}^s \sum_{j=1}^l A_{i,j} B_{i,j}$ . Therefore, we can easily solve the above linear regression model for  $p = 1, 2$  (the least square regression) and  $+\infty$ .

To relax the weak exogeneity assumption, we assume that the real data are contained in the following uncertainty set:

$$U = \left\{ [X; Y^T] : x_i = \bar{x}_i + \Delta x_i, \right.$$

$$\left. y_i = \bar{y}_i + \Delta y_i, i = 1, \dots, m, \left\| (\Delta y_i; \Delta x_i)_{i=1, \dots, m} \right\|_2 \leq \delta \right\}. \quad (5)$$

To minimize the worst-case residual error, we establish the following robust quadratic regression model:

$$\min_{\alpha, \beta, Q} \max_{[X; Y^T] \in U} \left\| \begin{pmatrix} y_1 - x_1^T Q x_1 - 2\alpha^T x_1 - \beta \\ \vdots \\ y_m - x_m^T Q x_m - 2\alpha^T x_m - \beta \end{pmatrix} \right\|_p. \quad (6)$$

From the computational perspective, although the robust linear regression problem (where the coefficients  $Q$  are set to zero) with a large variety of uncertainty sets can be efficiently solved, the robust quadratic regression problems are much more difficult. Actually for general uncertainty sets and least square criteria, even the inner maximization problem, which includes convex biquadratic polynomial as the objective function and general convex set as feasible set, is in general not solvable in polynomial run time. Next we will introduce some meaningful uncertainty sets and provide the corresponding tractable equivalences.

**2.2. Separable Ball Uncertainty Sets Model.** In this subsection, we consider the following separable ball uncertainty set:

$$U_s = U_1 \times U_2 \times \dots \times U_m, \quad (7)$$

where

$$U_i = \left\{ (x_i; y_i) \in R^{n+1} : x_i = \bar{x}_i + \Delta x_i, \right. \\ \left. y_i = \bar{y}_i + \Delta y_i, \left\| (\Delta y_i; \Delta x_i) \right\|_2 \leq \delta_i \right\}, \quad (8)$$

and  $\delta_i \geq 0$ . Thus the inner problem (IP) is of the following form (here we first consider square of the original objective function):

$$(IP) \max \sum_{i=1}^m (y_i - x_i^T Q x_i - 2\alpha^T x_i - \beta)^2 \\ \text{s.t. } (x_i; y_i) \in U_i, \quad i = 1, \dots, m. \quad (9)$$

Note that for the inner problem, the separable uncertainty set and the summation form of the objective function allow us to decompose it into  $m$  small scale subproblems with quadratic objective function and ball constraints. The quadratic objective function and constraints motivate us to use the following S-lemma to obtain an equivalent solvable reformulation.

**Lemma 1** (inhomogeneous version of S-lemma [8]). *Let  $A, B$  be symmetric matrices of the same size, and let the quadratic form  $x^T A x + 2a^T x + \beta$  be strictly positive at some point. Then the implication*

$$x^T A x + 2a^T x + \beta \geq 0 \implies x^T B x + 2b^T x + \beta \geq 0 \quad (10)$$

holds true if and only if

$$\exists \lambda \geq 0: \begin{bmatrix} B - \lambda A & (b - \lambda a)^T \\ b - \lambda a & \beta - \lambda \alpha \end{bmatrix} \succeq 0. \quad (11)$$

We can obtain the following equivalent semidefinite programming for the separable robust least square quadratic regression model.

**Proposition 2.** *The robust least square quadratic regression model with separable uncertainty set  $U_s$  is equivalent to the following semidefinite programming:*

$$\min_{\alpha, \beta, Q, v, u, r, t, \tau} v \\ \text{s.t. } \begin{pmatrix} u_i - t_i \delta_i^2 & & \\ & t_i & \\ & & t_i I_{n \times n} \end{pmatrix} + F \succeq 0, \quad i \in M_+, \\ \begin{pmatrix} u_i - \tau_i \delta_i^2 & & \\ & \tau_i & \\ & & \tau_i I_{n \times n} \end{pmatrix} - F \succeq 0, \quad i \in M_+, \\ r_i + u_i \geq 0, \quad r_i - u_i \geq 0, \quad i \in M_0, \\ r_i = \beta + \bar{x}_i^T Q \bar{x}_i + 2\alpha^T \bar{x}_i - \bar{y}_i, \quad i = 1, \dots, m, \\ \begin{pmatrix} v & u^T \\ u & v I_{m \times m} \end{pmatrix} \succeq 0, \\ t_i \geq 0, \tau_i \geq 0, u_i, r_i \in R, \quad i = 1, \dots, m, \\ v, \beta \in R, \alpha \in R^n, Q \in R^{n \times n}, \quad (12)$$

where

$$F = \begin{pmatrix} r_i & -\frac{1}{2} & (Q^T \bar{x}_i + \alpha)^T \\ -\frac{1}{2} & 0 & 0_{1 \times n} \\ Q^T \bar{x}_i + \alpha & 0_{n \times 1} & Q \end{pmatrix}, \quad (13)$$

$$M_+ = \{i : \delta_i > 0\} \cap \{1, \dots, m\},$$

$$M_0 = \{i : \delta_i = 0\} \cap \{1, \dots, m\}.$$

*Proof.* First consider the inner maximization subproblem. It is obvious that

$$\begin{aligned}
(\text{IP}_i) \quad & \max \quad (y_i - x_i^T Q x_i - 2\alpha^T x_i - \beta)^2 \\
& \text{s.t.} \quad (x_i; y_i) \in U_i, \\
\iff \min \quad & u_i^2 \\
& \text{s.t.} \quad u_i \geq |y_i - x_i^T Q x_i - 2\alpha^T x_i - \beta|, \quad \forall (x_i; y_i) \in U_i.
\end{aligned} \tag{14}$$

If  $\delta_i = 0$ , we have that

$$\begin{aligned}
& x_i^T Q x_i + 2\alpha^T x_i + \beta - y_i + u_i \geq 0, \\
& \forall (x_i; y_i) \in U_i \iff r_i + u_i \geq 0, \\
& x_i^T Q x_i + 2\alpha^T x_i + \beta - y_i - u_i \geq 0, \\
& \forall (x_i; y_i) \in U_i \iff r_i - u_i \geq 0,
\end{aligned} \tag{15}$$

where  $r_i = \bar{x}_i^T Q \bar{x}_i + 2\alpha^T \bar{x}_i + \beta - \bar{y}_i$ .

If  $\delta_i > 0$ , we can utilize the S-lemma as follows:

$$\begin{aligned}
& x_i^T Q x_i + 2\alpha^T x_i + \beta - y_i + u_i \geq 0, \quad \forall (x_i; y_i) \in U_i, \\
\iff \Delta x_i^T Q \Delta x_i + 2(Q^T \bar{x}_i + \alpha)^T \Delta x_i - \Delta y_i + r_i + u_i \geq 0, \\
& \forall \|(\Delta y_i; \Delta x_i)\|_2 \leq \delta_i,
\end{aligned}$$

$$\begin{aligned}
\iff & \begin{pmatrix} 1 \\ \Delta y_i \\ \Delta x_i \end{pmatrix}^T \begin{pmatrix} u_i + r_i & -\frac{1}{2} (Q^T \bar{x}_i + \alpha)^T \\ -\frac{1}{2} & 0 & 0_{1 \times n} \\ Q^T \bar{x}_i + \alpha & 0_{n \times 1} & Q \end{pmatrix} \\
& \times \begin{pmatrix} 1 \\ \Delta y_i \\ \Delta x_i \end{pmatrix} \geq 0, \quad \forall \|(\Delta y_i; \Delta x_i)\|_2 \leq \delta_i,
\end{aligned}$$

$$\begin{aligned}
\iff \exists t_i \geq 0, \quad & \text{s.t.} \begin{pmatrix} u_i + r_i - t_i \delta_i^2 & -\frac{1}{2} (Q^T \bar{x}_i + \alpha)^T \\ -\frac{1}{2} & t_i & 0_{1 \times n} \\ Q^T \bar{x}_i + \alpha & 0_{n \times 1} & Q + t_i I_{n \times n} \end{pmatrix} \\
& \geq 0.
\end{aligned} \tag{16}$$

Note that in the last step, if  $\delta_i^2 > 0$ , then there exists  $(\Delta y_i; \Delta x_i) = (0; 0)$  such that quadratic form  $\delta_i^2 - \|(\Delta y_i; \Delta x_i)\|_2^2$  is strictly positive; thus the condition of S-lemma holds truly. Similarly we have that

$$\begin{aligned}
& x_i^T Q x_i + 2\alpha^T x_i + \beta - y_i - u_i \leq 0, \quad \forall (x_i; y_i) \in U_i, \\
\iff \Delta x_i^T Q \Delta x_i + 2(Q^T \bar{x}_i + \alpha)^T \Delta x_i - \Delta y_i + r_i - u_i \leq 0, \\
& \forall \|(\Delta y_i; \Delta x_i)\|_2 \leq \delta_i,
\end{aligned}$$

$$\begin{aligned}
\iff \exists \tau_i \geq 0, \quad & \text{s.t.} \begin{pmatrix} u_i - r_i - \tau_i \delta_i^2 & \frac{1}{2} & -(Q^T \bar{x}_i + \alpha)^T \\ \frac{1}{2} & \tau_i & 0_{1 \times n} \\ -Q^T \bar{x}_i - \alpha & 0_{n \times 1} & \tau_i I_{n \times n} - Q \end{pmatrix} \\
& \geq 0.
\end{aligned} \tag{17}$$

Thus the inner maximization problem is equivalent to the following semi-definite programming:

$$\begin{aligned}
(\text{IP}) \quad & \min_{u_i, r_i, t_i, \tau_i} \sum_{i=1}^m u_i^2 \\
& \text{s.t.} \begin{pmatrix} u_i + r_i - t_i \delta_i^2 & -\frac{1}{2} (Q^T \bar{x}_i + \alpha)^T \\ -\frac{1}{2} & t_i & 0_{1 \times n} \\ Q^T \bar{x}_i + \alpha & 0_{n \times 1} & Q + t_i I_{n \times n} \end{pmatrix} \geq 0, \\
& i \in M_+,
\end{aligned}$$

$$\begin{aligned}
& \begin{pmatrix} u_i - r_i - \tau_i \delta_i^2 & \frac{1}{2} & -(Q^T \bar{x}_i + \alpha)^T \\ \frac{1}{2} & \tau_i & 0_{1 \times n} \\ -Q^T \bar{x}_i - \alpha & 0_{n \times 1} & \tau_i I_{n \times n} - Q \end{pmatrix} \geq 0, \\
& i \in M_+,
\end{aligned}$$

$$r_i + u_i \geq 0, \quad r_i - u_i \geq 0 \quad i \in M_0,$$

$$r_i = \beta + \bar{x}_i^T Q \bar{x}_i + 2\alpha^T \bar{x}_i - \bar{y}_i \quad i = 1, \dots, m,$$

$$t_i \geq 0, \tau_i \geq 0, u_i, r_i \in R \quad i = 1, \dots, m.$$

(18)

Note that based on the Schur complement lemma, the second-order cone constraint  $v \geq \sqrt{\sum_{i=1}^m u_i^2}$  can also be formalized as the following semi-definite constraint:

$$\begin{pmatrix} v & u^T \\ u & v I_{m \times m} \end{pmatrix} \geq 0. \tag{19}$$

Thus we complete the proof by embedding the equivalent semi-definite programming into the outer problem.  $\square$

Due to the advance of interior algorithms for conic programming, the above semidefinite programming can be efficiently solved in polynomial run time. There are several efficient and free software packages for solving the semidefinite programming, such as the SDPT3 [19]. Next we make several extensions based on the separable robust least square quadratic regression model.

**2.3. Ellipsoid Uncertainty Set and More Norm Criterion.** The above result on standard ball uncertainty set can be further extended to that on the following general ellipsoid uncertainty set:

$$U'_s = U'_1 \times U'_2 \times \cdots \times U'_m,$$

$$U'_i = \{(x_i; y_i) \in R^{n+1} : x_i = \bar{x}_i + \Delta x_i, \quad (20)$$

$$y_i = \bar{y}_i + \Delta y_i, \|P_i(\Delta y_i; \Delta x_i)\|_2 \leq \delta_i\},$$

where  $P_i \in R^{k \times (n+1)}$ . Linear transformation operator  $P_i$  allows us to impose more restrictions on the uncertainty set. For example, if we choose the diagonal matrix  $P_i = \text{Diag}\{\sigma_1, \dots, \sigma_{n+1}\}$ , we can put different weights on deviation of components of  $(x_i; y_i)$ ; general matrix can further restrict the correlated deviation of different components.

To obtain the corresponding reformulation, we only need to modify the first two constraints based on the S-lemma as follows:

$$\begin{pmatrix} u_i - t_i \delta_i^2 & & \\ & t_i P_i^T P_i & \\ & & \end{pmatrix} + F \geq 0 \quad i \in M_+,$$

$$\begin{pmatrix} u_i - \tau_i \delta_i^2 & & \\ & \tau_i P_i^T P_i & \\ & & \end{pmatrix} - F \geq 0 \quad i \in M_+. \quad (21)$$

We further consider the robust quadratic regression models with  $l_\infty$ -norm and  $l_1$ -norm criterion. Note that for  $l_\infty$ -norm criteria, the inner maximization problem is of the following form:

$$\begin{aligned} & \max_{1 \leq i \leq m} \max_{(x_i; y_i) \in U_i} \{y_i - x_i^T Q x_i - 2\alpha^T x_i - \beta\}, \\ \iff & \min \{u : u \geq |y_i - x_i^T Q x_i - 2\alpha^T x_i - \beta|, \quad (22) \\ & \forall (x_i; y_i) \in U_i, 1 \leq i \leq m\}. \end{aligned}$$

And for  $l_1$ -norm criteria, we have the following equivalent reformulation:

$$\begin{aligned} & \max_{1 \leq i \leq m} \left\{ \sum_{i=1}^m |y_i - x_i^T Q x_i - 2\alpha^T x_i - \beta| \right. \\ & \left. : (x_i; y_i) \in U_i, 1 \leq i \leq m \right\}, \\ \iff & \min \left\{ \sum_{i=1}^m u_i : u_i \geq |y_i - x_i^T Q x_i - 2\alpha^T x_i - \beta|, \right. \\ & \left. \forall (x_i; y_i) \in U_i, 1 \leq i \leq m \right\}. \quad (23) \end{aligned}$$

Using the similar approach as in Proposition 2, both can be further reformulated as semi-definite programming.

**Proposition 3.** The separable robust quadratic regression model under  $l_\infty$ -norm and  $l_1$ -norm criteria are equivalent to the following semidefinite programming, respectively:

$$\begin{aligned} & (l_\infty\text{-norm}) \min_{\alpha, \beta, Q, u, r, t, \tau} u \\ & \text{s.t.} \quad \begin{pmatrix} u - t_i \delta_i^2 & & \\ & t_i & \\ & & t_i I_{n \times n} \end{pmatrix} + F \geq 0, \quad i \in M_+, \\ & \quad \begin{pmatrix} u - \tau_i \delta_i^2 & & \\ & \tau_i & \\ & & \tau_i I_{n \times n} \end{pmatrix} - F \geq 0, \quad i \in M_+, \\ & \quad r_i + u \geq 0, \quad r_i - u \geq 0, \quad i \in M_0, \\ & \quad r_i = \beta + \bar{x}_i^T Q \bar{x}_i + 2\alpha^T \bar{x}_i - \bar{y}_i, \quad i = 1, \dots, m, \\ & \quad t_i \geq 0, \tau_i \geq 0, r_i \in R, \quad i = 1, \dots, m, \\ & \quad u, \beta \in R, \alpha \in R^n, Q \in R^{n \times n}, \end{aligned}$$

$$\begin{aligned} & (l_1\text{-norm}) \min_{\alpha, \beta, Q, u, r, t, \tau} \sum_{i=1}^m u_i \\ & \text{s.t.} \quad \begin{pmatrix} u_i - t_i \delta_i^2 & & \\ & t_i & \\ & & t_i I_{n \times n} \end{pmatrix} + F \geq 0, \quad i \in M_+, \\ & \quad \begin{pmatrix} u_i - \tau_i \delta_i^2 & & \\ & \tau_i & \\ & & \tau_i I_{n \times n} \end{pmatrix} - F \geq 0, \quad i \in M_+, \\ & \quad r_i + u_i \geq 0, \quad r_i - u_i \geq 0, \quad i \in M_0, \\ & \quad r_i = \beta + \bar{x}_i^T Q \bar{x}_i + 2\alpha^T \bar{x}_i - \bar{y}_i, \\ & \quad i = 1, \dots, m, \\ & \quad t_i \geq 0, \tau_i \geq 0, u_i, r_i \in R \quad i = 1, \dots, m, \\ & \quad \beta \in R, \alpha \in R^n, Q \in R^{n \times n}, \end{aligned} \quad (24)$$

where

$$F = \begin{pmatrix} r_i & -\frac{1}{2} & (Q^T \bar{x}_i + \alpha)^T \\ -\frac{1}{2} & 0 & 0_{1 \times n} \\ Q^T \bar{x}_i + \alpha & 0_{n \times 1} & Q \end{pmatrix}. \quad (25)$$

### 3. Robust Energy-Growth Regression Models

Studies have been reported on the causal relationship between economic growth and energy consumption. In this section, we try to apply the proposed robust quadratic regression model to the energy-growth problem.

The seminal paper of J. Kraft and A. Kraft [20] first studies the casual relationship for USA. In a recent survey, Ilhan [21] categorizes the casual relationships into four types: no causality, unidirectional causality running from economic growth

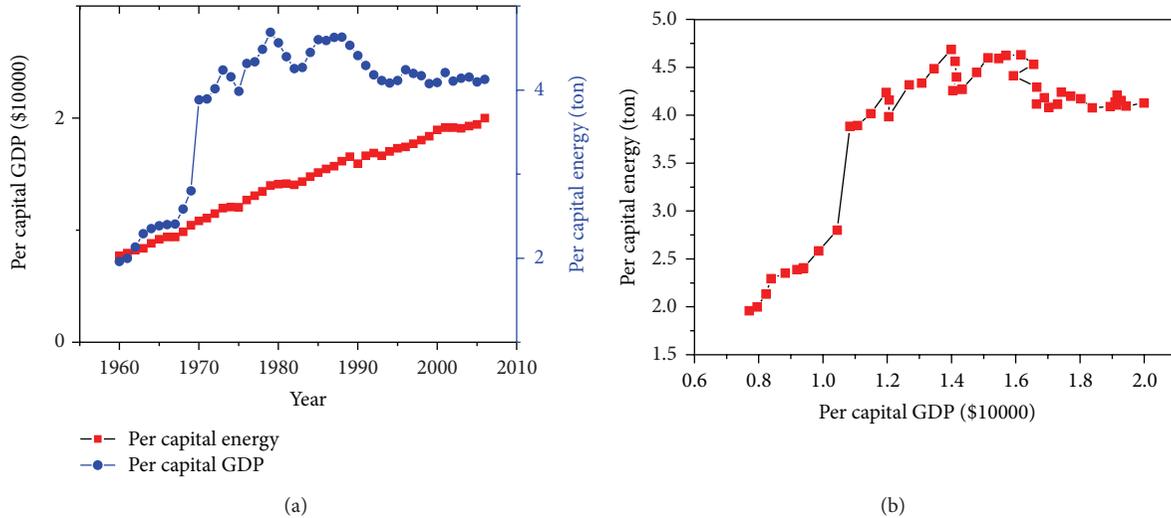


FIGURE 1: Germany data from 1960 to 2006.

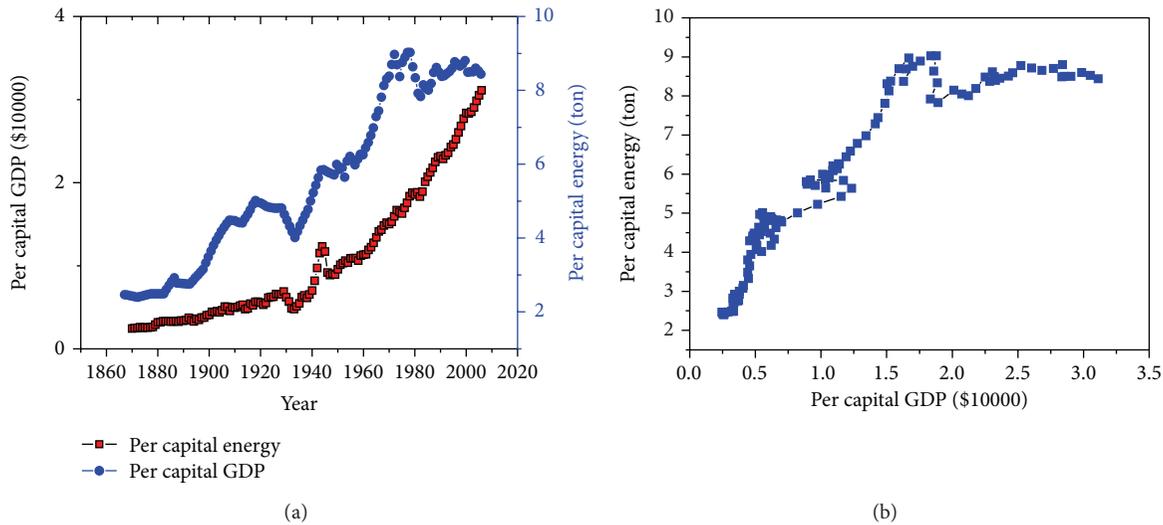


FIGURE 2: USA data from 1870 to 2006.

to energy consumption, the reverse case, and the bidirectional causality. Note that the resulted relationships depend on the selected data and analysis approaches. Sometimes the results obtained from different approaches conflict with each other when even using the data from the same country. For example, using the Toda-Yamamoto causality test method, Bowden and Payne [22] show that energy consumption plays an important role in economic growth in USA based on history data from 1949 to 2006 while using the same method Soytaş and Sari [23] find that no causality exists between them based on USA data from 1960 to 2006. On the other hand, based on the same USA's data from 1947 to 1990, Cheng [24] and Stern [25] conclude different causalities by utilizing different analyzing approaches.

Unlike the previous energy-growth studies, we attempt to provide a long-run stationary regression model between the per capita GDP ( $G$ ) and per capita energy consumption

( $EC$ ). The underlying assumption of our model is similar to the traditional "conservation hypothesis" that means that an increase in real GDP will cause an increase in energy consumption [21]. The "per capita" perspective provides us with a new insight on the causality and new regression models. Figures 1 and 2 demonstrate the relationship between per capita energy consumption and per capita GDP in USA and Germany respectively. From the subfigures on the left hand side, we can see that in both countries there is a gradual increase in economy while the per capita energy consumption may decrease after reaching a certain level; the subfigures on the right hand side inspire us to establish a nonlinear regression model to characterize the relationship.

To eliminate effect of the imprecise statistics data, we employ the proposed robust quadratic regression model and put different weights on the residual errors at different time

TABLE 1: LS-CQR and LS-RQR models with different  $\epsilon$ .

Model	Q	$\alpha$	$\beta$	Err	T (s)
CQR $\epsilon = 0.00$	-4.254	6.721	-6.099	1.688	0.000
RQR $\epsilon = 0.01$	-3.899	6.225	-5.433	1.621	0.500
RQR $\epsilon = 0.02$	-3.690	5.938	-5.063	1.663	0.500
RQR $\epsilon = 0.03$	-3.423	5.561	-4.564	1.735	0.500
RQR $\epsilon = 0.04$	-2.900	4.755	-3.363	2.029	0.516
RQR $\epsilon = 0.05$	-2.243	3.719	-1.817	2.574	0.484

points. Specifically we establish the following weighted robust quadratic regression model:

$$\min_{\alpha, \beta, q} \max_{(G_t, EC_t) \in U_t^\epsilon} \left( \sum_{t=1}^T (w_t (EC_t - qG_t^2 - 2\alpha G_t - \beta))^p \right)^{1/p}, \quad (26)$$

where the weight factor  $w_t \in [0, 1]$  represents the relative importance of the predicted residual error in the  $t$ th year. We could set  $w_t = 0$  for the abnormal data point and set  $w_t$  as an increase function of  $t$  to emphasize the importance of recent data. The uncertainty set is defined as

$$U_t^\epsilon = \{(G_t, EC_t) : \|(G_t - \bar{G}_t), (EC_t - \bar{EC}_t)\|_2 \leq \delta_t\}, \quad (27)$$

where  $\delta_t = \epsilon \sqrt{\bar{G}_t^2 + \bar{EC}_t^2}$ . Parameter  $\epsilon$  controls the relative amplitude of the fluctuation in observed data.

The weighted robust quadratic regression model can be summarized as follows.

- (1) Solve the classical quadratic regression model using the nominal values  $(\bar{G}_t, \bar{EC}_t)_{t=1}^T$ .
- (2) Based on the quadratic regression, remove the data with the first  $k$  largest residual errors and set weights value  $w_t$ .
- (3) Solve the equivalent semi-definite programming problem and return the final weighted robust quadratic regression model.

#### 4. Numerical Experiments

In this section, we verify the effectiveness of the proposed robust quadratic regression models on several data sets. The equivalent semi-definite programming problem is solved by the SDPT3 solver [19]. Numerical experiments are implemented using MATLAB 7.7.0 and run on Intel(R) Core(TM)2 CPU E7400.

First we test the proposed robust least square quadratic regression (LS-RQR) model with Germany data from 1960 to 2006. As previously discussed, after the preliminary quadratic regression analysis, we will remove the data with the first  $k$  largest residual errors, where  $k = 3\% \times$  data size. Then for the rest of data, we establish the classical least square quadratic regression (LS-CQR) and LS-RQR models, respectively.

Table 1 lists the computation results for LS-CQR and LS-RQR with a series of  $\epsilon$  values. The listed Err value

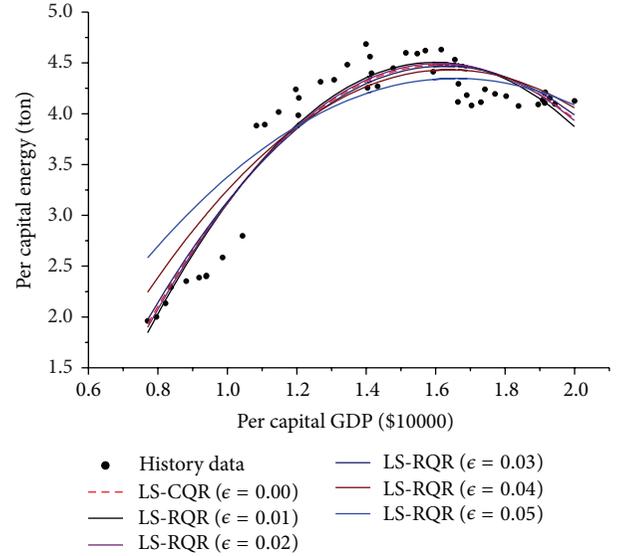
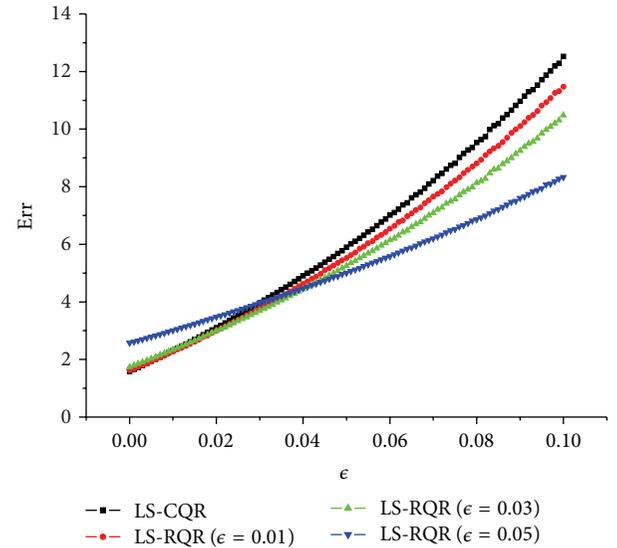


FIGURE 3: LS-CQR and LS-RQR models on Germany data.


 FIGURE 4: Mean square error of LS-CQR and LS-RQR models when  $\epsilon$  varies.

represents the mean square error from the nominal value, and  $T$  represents the run time for solving the optimization problem. It is seen that the resulted robust model exhibits smaller absolute values of  $Q$ ,  $\alpha$ , and  $\beta$  with the increase of  $\epsilon$  value; that is, the regression curve is more flat as the model

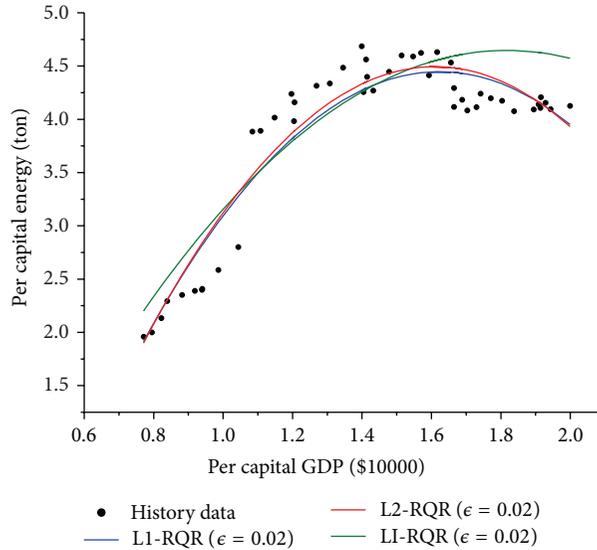


FIGURE 5: RQR models under  $l_1$ -,  $l_2$ - and  $l_\infty$ -norm criteria.

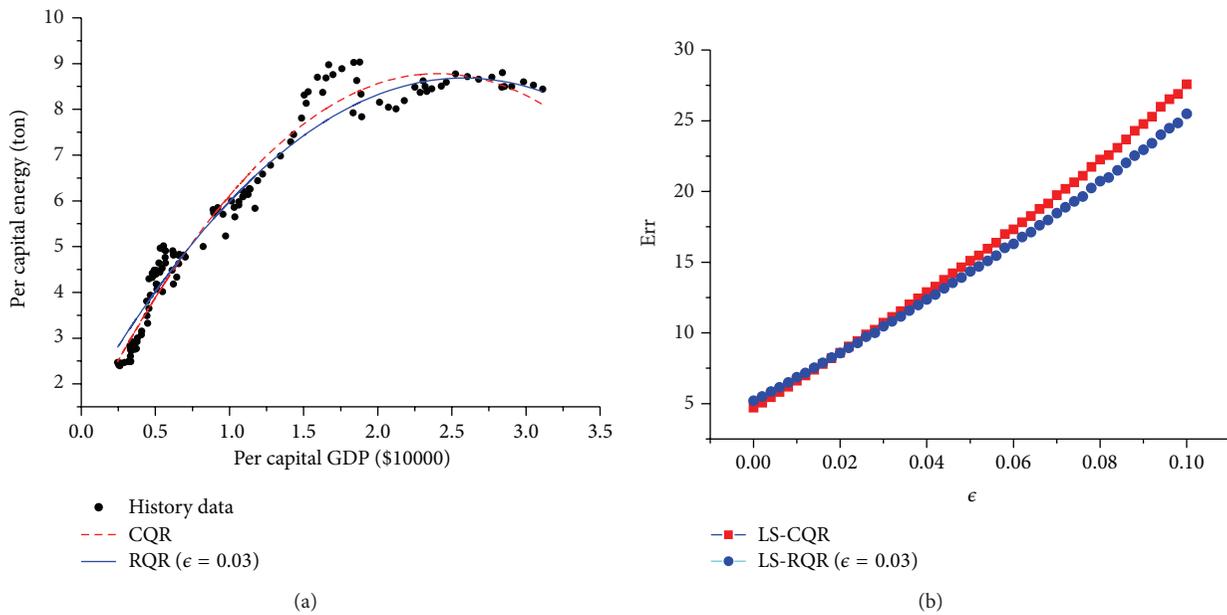


FIGURE 6: USA data from 1870 to 2006.

parameters are less precise. It is obvious that one drawback of the robust model is that the mean square error will increase as uncertainty increases. Figure 3 plots the regression curves for different models and also supports our analysis of the effect of increasing data uncertainty on robust regression.

To demonstrate the effectiveness of the robust models, we test the worst-case performance of the resulted models when  $\epsilon$  varies from 0 to 0.1. Specifically, for each  $\epsilon$  value, we randomly generate 500 groups of data from the defined uncertainty set  $U_t^\epsilon$  and then calculate the maximal residual error at each data point. Figure 4 plots the worst-case error of LS-CQR model and LS-RQR models with  $\epsilon = 0.01, 0.03$ , and 0.05. It is seen that the error of LS-CQR model increases

rapidly, and LS-RQR with  $\epsilon = 0.05$  has the most flat error curve. Figure 4 also indicates that it is critical to accurately estimate the variability of the data and set proper value for  $\epsilon$ . In our case, we recommend LS-RQR with  $\epsilon = 0.03$  that is almost always better than the traditional LS-CQR model.

Next we test the proposed RQR models under  $l_1$  (LI-RQR) and  $l_\infty$ - (LI-RQR) norm criteria on the same data set. Figure 5 plots the corresponding regression curves for the same uncertainty set  $\epsilon = 0.02$ . For the same  $\epsilon$  value, LI-RQR model can be considered as the most robust one, and LI-RQR and L2-RQR models are similar. It is noticeable that it contradicts with the traditional robust regression terms. For example, [26] refers to the  $l_1$ -norm regression as the robust

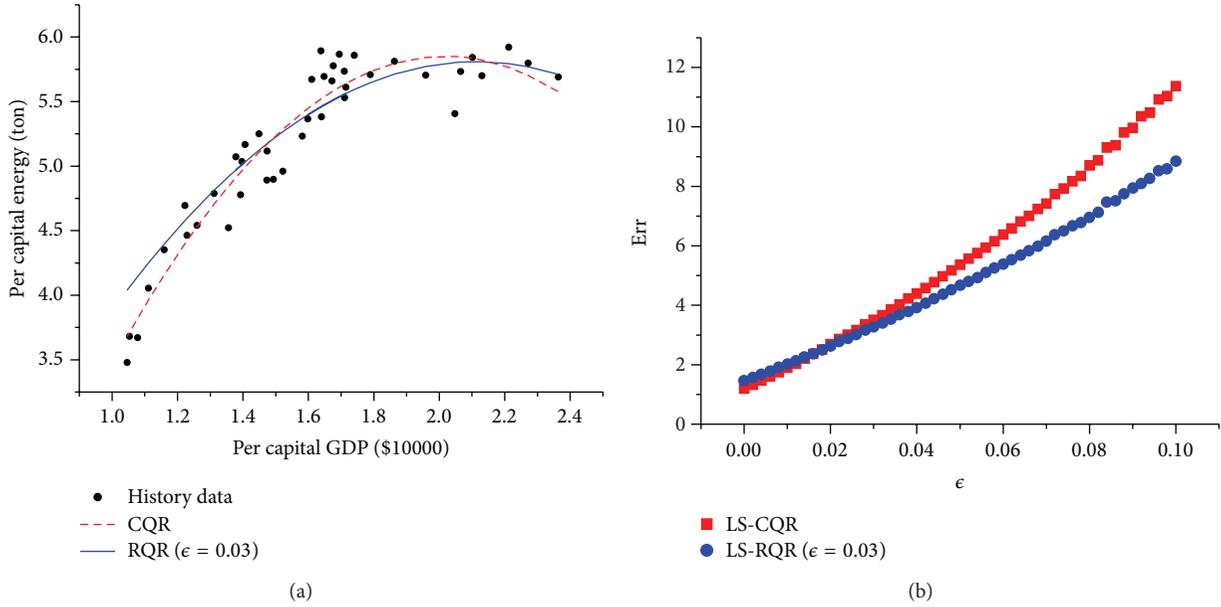


FIGURE 7: Switzerland data from 1965 to 2006.

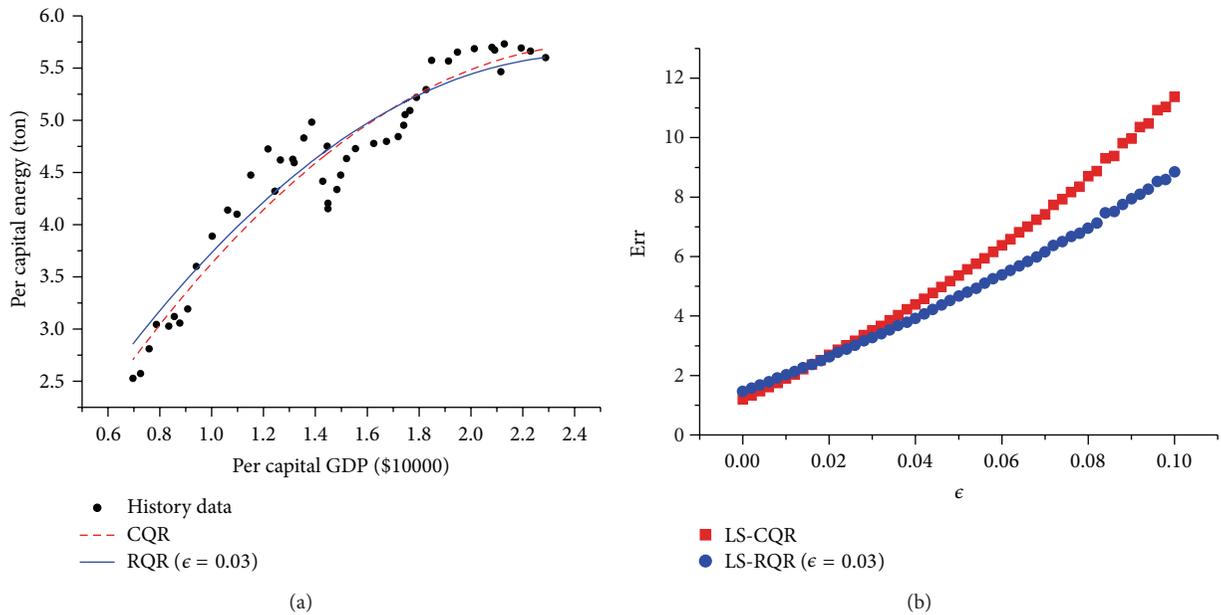


FIGURE 8: Belgium data from 1960 to 2006.

regression model in the sense that the corresponding model is insensitive to the large residual errors (corresponding to the outliers). However, after removing the possible abnormal data points, here we try to make our regression analysis insensitive to the worst-case residual errors at each data point.

Finally we apply the proposed RQR model on more data sets, including USA data from 1870 to 2006, Switzerland data from 1965 to 2006, and Belgium data from 1960 to 2006. Figures 6, 7, and 8 give the resulted regression models and the worst-case residual errors for different  $\epsilon$  values. It is seen that the proposed RQR models still almost always outperform the

CQR model, especially for large uncertainty sets. Based on the robust quadratic regression models, these three countries reach the highest per capita energy consumption points at per capita GDP value around 23,000 while the peak values vary from 5.7 to 8.5 Ton.

### 5. Conclusions and Future Works

In this paper, we studied the multivariate quadratic regression model with imprecise statistic data. Unlike the traditional robust statistic approaches that focus on the detection of

the outliers and the elimination of the effects, we employed the recently developed robust optimization framework and uncertainty set theory.

In particular, we first extended the existing robust linear regression results to the robust least square quadratic regression model with the separable ball uncertainty set. The specific form of the uncertainty set allowed us to use the well-known S-lemma and give the tractable equivalent semidefinite programming. We further generalized the result to robust models under  $l_1$ - and  $l_\infty$ -norm criteria with general ellipsoid uncertainty sets. Next the proposed robust models were applied to the energy-growth problem. Under the classical conservation hypothesis, we employed the traditional quadratic regression model to remove the abnormal data and established a robust quadratic regression model for the per capital GDP and per capital energy consumption. Finally the proposed models were tested on the history data of Germany, USA, Switzerland, and Belgium. From the numerical experiments, we found that (1) the amplitude of the uncertainty perturbation  $\delta$  plays a critical role on the robust models; (2) with the increase of  $\delta$ , the robust model has a more flat curve; (3) for the same  $\delta$  value, compared with  $l_1$ - and  $l_2$ -norm models,  $l_\infty$ -norm model is the most robust one; (4) as expected, the robust approach provides a serial robust regression models that can reduce the worst-case residual errors when the observed data contain noise.

For further research, robust polynomial (nonlinear) regression models are interesting in their own right. Although we may always reduce them to the linear regression model with polynomially (or nonlinearly) transformed uncertainty data set, it is still worth studying whether the resulted regression models are solvable for quadratic regression with coupled uncertainty sets.

## Acknowledgment

This work was supported by Geological Survey Project of China (nos. 1212010881801, 1212011120995).

## References

- [1] Z. Griliches and V. Ringstad, "Errors-in-the-variables bias in nonlinear contexts," *Econometrica*, vol. 38, no. 2, pp. 368–370, 1970.
- [2] W. A. Fuller, *Measurement Error Models*, John Wiley & Sons, New York, NY, USA, 1987.
- [3] T. Erickson and T. M. Whited, "Two-step GMM estimation of the errors-in-variables model using high-order moments," *Econometric Theory*, vol. 18, no. 3, pp. 776–799, 2002.
- [4] P. J. Cornbleet and N. Gochman, "Incorrect least-squares regression coefficients," *Clinical Chemistry*, vol. 25, no. 3, pp. 432–438, 1979.
- [5] J. W. Gillard, "An historical overview of linear regression with errors in both variables," Tech. Rep., Cardiff University School of Mathematics, Cardiff, UK, 2006.
- [6] L. El Ghaoui and H. Lebret, "Robust solutions to least-squares problems with uncertain data," *SIAM Journal on Matrix Analysis and Applications*, vol. 18, no. 4, pp. 1035–1064, 1997.
- [7] P. K. Shivaswamy, C. Bhattacharyya, and A. J. Smola, "Second order cone programming approaches for handling missing and uncertain data," *Journal of Machine Learning Research*, vol. 7, pp. 1283–1314, 2006.
- [8] A. Ben-Tal, L. El Ghaoui, and A. Nemirovski, *Robust Optimization*, Princeton University Press, Princeton, NJ, USA, 2009.
- [9] T. B. Trafalis and R. C. Gilbert, "Robust classification and regression using support vector machines," *European Journal of Operational Research*, vol. 173, no. 3, pp. 893–909, 2006.
- [10] H. Xu, C. Caramanis, and S. Mannor, "Robustness and regularization of support vector machines," *Journal of Machine Learning Research*, vol. 10, pp. 1485–1510, 2009.
- [11] T. B. Trafalis and R. C. Gilbert, "Robust support vector machines for classification and computational issues," *Optimization Methods & Software*, vol. 22, no. 1, pp. 187–198, 2007.
- [12] P. J. Huber, *Robust Statistics*, John Wiley & Sons, New York, NY, USA, 1981.
- [13] J. L. Wu and P. C. Chang, "A trend-based segmentation method and the support vector regression for financial time series forecasting," *Mathematical Problems in Engineering*, vol. 2012, Article ID 615152, 20 pages, 2012.
- [14] D. X. She and X. H. Yang, "A new adaptive local linear prediction method and its application in hydrological time Series," *Mathematical Problems in Engineering*, vol. 2010, Article ID 205438, 15 pages, 2010.
- [15] Z. Liu, "Chaotic time series analysis," *Mathematical Problems in Engineering*, vol. 2010, Article ID 720190, 31 pages, 2010.
- [16] M. Li, "Fractal time series: a tutorial review," *Mathematical Problems in Engineering*, vol. 2010, Article ID 157264, 26 pages, 2010.
- [17] T. Farooq, A. Guergachi, and S. Krishnan, "Knowledge-based green's kernel for support vector regression," *Mathematical Problems in Engineering*, vol. 2010, Article ID 378652, 16 pages, 2010.
- [18] I. Pólik and T. Terlaky, "A survey of the S-lemma," *SIAM Review*, vol. 49, no. 3, pp. 371–418, 2007.
- [19] K. C. Toh, R. H. Tütüñü, and M. J. Todd, "On the implementation and usage of SDPT3Ca Matlab software package for semidefinitequadratic-linear programming," version 4. 0, 2006, <http://ecommons.library.cornell.edu/handle/1813/15133>.
- [20] J. Kraft and A. Kraft, "On the relationship between energy and GNP," *Journal of Energy and Development*, vol. 3, no. 2, pp. 401–403, 1978.
- [21] O. Ilhan, "A literature survey on energy growth nexus," *Energy Policy*, vol. 38, pp. 340–349, 2010.
- [22] N. Bowden and J. E. Payne, "The causal relationship between US energy consumption and real output: a disaggregated analysis," *Journal of Policy Modeling*, vol. 31, no. 2, pp. 180–188, 2009.
- [23] U. Soytaş and R. Sari, "Energy consumption, economic growth, and carbon emissions: challenges faced by an EU candidate member," *Ecological Economics*, vol. 68, no. 6, pp. 1667–1675, 2009.
- [24] B. Cheng, "An investigation of cointegration and causality between energy consumption and economic growth," *Journal of Energy Development*, vol. 21, no. 1, pp. 73–84, 1995.
- [25] D. I. Stern, "Energy and economic growth in the USA: a multivariate approach," *Energy Economics*, vol. 15, no. 2, pp. 137–150, 1993.
- [26] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, Cambridge, UK, 2004.

## Research Article

# LGMS-FOA: An Improved Fruit Fly Optimization Algorithm for Solving Optimization Problems

**Dan Shan, GuoHua Cao, and HongJiang Dong**

*School of Economics and Business Administration, Chongqing University, Chongqing 400030, China*

Correspondence should be addressed to GuoHua Cao; [caoguohua@cqu.edu.cn](mailto:caoguohua@cqu.edu.cn)

Received 16 May 2013; Revised 1 August 2013; Accepted 18 August 2013

Academic Editor: Yudong Zhang

Copyright © 2013 Dan Shan et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Recently, a new fruit fly optimization algorithm (FOA) is proposed to solve optimization problems. In this paper, we empirically study the performance of FOA. Six different nonlinear functions are selected as testing functions. The experimental results illustrate that FOA cannot solve complex optimization problems effectively. In order to enhance the performance of FOA, an improved FOA (named LGMS-FOA) is proposed. Simulation results and comparisons of LGMS-FOA with FOA and other metaheuristics show that LGMS-FOA can greatly enhance the searching efficiency and greatly improve the searching quality.

## 1. Introduction

Optimization problems are used extensively in science, engineering, and finance. How to solve these problems has always been a concern of researchers. In the past decade, stochastic optimization algorithms have been used to solve these problems due to their flexibility for finding solutions. These algorithms include genetic algorithm (GA) [1–3], simulated annealing (SA) [4, 5], ant colony optimization algorithm [5, 6], and particle swarm optimization algorithm (PSO) [7–9]. However, the common disadvantages of these stochastic algorithms are complicated computational processes and difficulty of understanding for beginners.

Recently, a new stochastic optimization technique, fruit fly optimization algorithm (FOA), is proposed by Pan [10]. Its development is based on the food finding behavior of the fruit fly. Compared with other stochastic algorithms, FOA has the advantages of being easy to understand and a simple computational process. As a novel optimization algorithm, FOA has gained much attention [10, 11] in recent years.

In order to study the performance of FOA, six famous nonlinear functions are selected as testing functions. Simulation results illustrate that FOA cannot solve the complex optimization problems effectively. Analysis of FOA shows that FOA includes a nonlinear generation mechanism of candidate solution. This mechanism is abbreviated as NGMS

and has some disadvantages which limit the performance of FOA. In order to enhance the performance of FOA, NGMS is first replaced with a linear generation mechanism of candidate solution (abbreviated as LGMS), and then a LGMS-based improved FOA (abbreviated as LGMS-FOA) is proposed. Simulation results and comparisons of LGMS-FOA with FOA and other metaheuristics show that LGMS-FOA is more effective and reliable.

The rest of this paper is organized as follows. Section 2 introduces FOA. Section 3 introduces LGMS-FOA. Section 4 provides comparisons of FOA with LGMS-FOA and other metaheuristics. Section 5 concludes this paper.

## 2. FOA

*2.1. Overview of FOA.* FOA is a new method for finding global optimization based on the food finding behavior of the fruit fly. The fruit fly is superior to other species in vision and osphresis (as illustrated in Figure 1). The food finding process of fruit fly has two steps: firstly, it smells the food source by using osphresis organ and flies towards that direction; then, after it gets close to the food location, it can also use its sensitive vision to find food and fruit flies' flocking location and flies towards that direction. Figure 2 shows the food finding iterative process of fruit fly swarm.

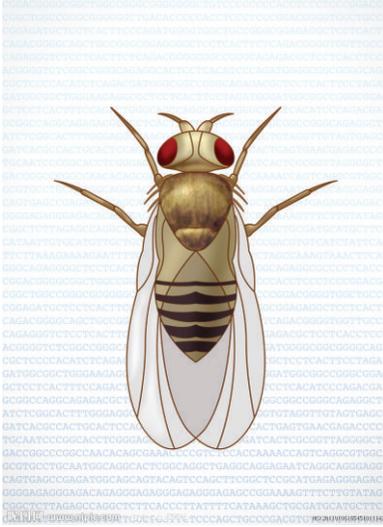


FIGURE 1: Fruit fly.

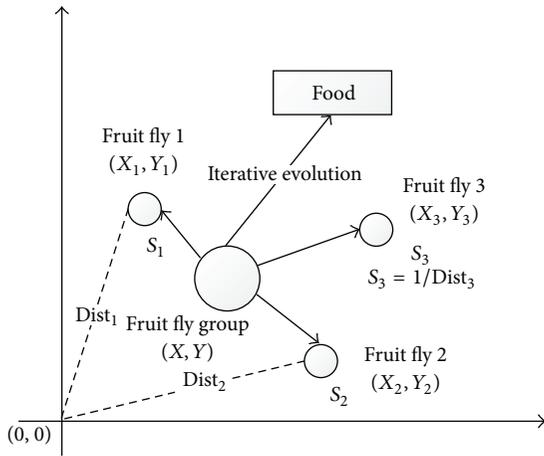


FIGURE 2: Food finding iterative process of fruit fly swarm.

Based on the food finding characteristics of fruit fly swarm, the whole procedure of FOA is described as follows.

*Step 1* (parameters initialization). The main parameters of FOA are the maximum iteration number (maxgen), the population size (sizepop), the random initialization fruit fly swarm location range (LR), and the random fly direction and distance zone of fruit fly (FR).

*Step 2*. Nonlinear generation mechanism of candidate solution:

*Step 2.1*. Initial fruit fly swarm location,

$$\begin{aligned} x\_axis &= \text{rand}(\text{LR}), \\ y\_axis &= \text{rand}(\text{LR}). \end{aligned} \quad (1)$$

*Step 2.2*. Give the random direction and distance for food finding of an individual fruit fly using osphresis:

$$\begin{aligned} x_i &= x\_axis + \text{rand}(\text{FR}), \\ y_i &= y\_axis + \text{rand}(\text{FR}). \end{aligned} \quad (2)$$

*Step 2.3*. Calculate the distance of food location to the origin:

$$\text{Dist}_i = \sqrt{x_i^2 + y_i^2}. \quad (3)$$

*Step 2.4*. Calculate the smell concentration judgment value ( $S_i$ ):

$$S_i = \frac{1}{\text{Dist}_i}. \quad (4)$$

*Remark 1*. In fact,  $S_i$  is a candidate solution in the domain.

*Remark 2*. According to (1)–(4)

$$\begin{aligned} S_i &= \frac{1}{\text{Dist}_i} \\ &= \frac{1}{\sqrt{(x\_axis + \text{rand}(\text{FR}))^2 + (y\_axis + \text{rand}(\text{FR}))^2}}. \end{aligned} \quad (5)$$

Equation (5) is called NGMS.

*Step 3*. Calculate the smell concentration ( $\text{Smell}_i$ ) of the individual fruit fly location by inputting the smell concentration judgment value ( $S_i$ ) into the smell concentration judgment function (also called *objective function*):

$$\text{Smell}_i = \text{objective function}(S_i). \quad (6)$$

*Step 4*. Find out the fruit fly with maximal smell concentration among the fruit fly swarm:

$$[\text{Smell best best index}] = \max(\text{Smell}). \quad (7)$$

*Step 5*. Keep the maximal concentration value and  $x$ ,  $y$  coordinate. Then, the fruit fly swarm flies towards that location by using vision:

$$\begin{aligned} \text{Smell best} &= \text{best smell}, \\ x\_axis &= x(\text{best index}), \\ y\_axis &= y(\text{best index}). \end{aligned} \quad (8)$$

*Step 6*. Enter iterative optimization to repeat the implementation of Steps 2–5. When the smell concentration is not superior to the previous iterative smell concentration any more or the iterative number reaches the maximal iteration number, the circulation stops.

*2.2. Computational Experiments of FOA*. In order to study the performance of FOA, six different nonlinear functions are selected as testing functions [12]. Table 1 shows the name, dimension, optimal solution, and extreme point of them. For details refer to Appendix A.

TABLE 1: Testing function.

$f$	Function name	Dimension	Optimal solution	Extreme point
$f_{GP}$	Goldstein and Price	2	3	nonzero point
$f_{SH}$	Shubert	2	-186.7309	nonzero point
$f_{BR}$	Branin RCOC	2	0.397887	nonzero point
$f_{RA}$	Rastrigin	2	-2	zero point
$f_{SP30}$	Sphere	30	0	zero point
$f_{SP50}$	Sphere	50	0	zero point

TABLE 2: Fixed-iteration results of FOA with different LR and FR.

Parameters of FOA		PS (%)						Average PS (%)
FR	LR	$f_{GP}$	$f_{SH}$	$f_{BR}$	$f_{RA}$	$f_{SP30}$	$f_{SP50}$	
[-1, 1]	[-1, 1]	0	0	0	0	0	0	0
[-1, 1]	[-5, 5]	0	0	0	0	0	0	0
[-1, 1]	[-10, 10]	0	0	0	0	0	0	0
[-1, 1]	[-20, 20]	0	0	0	0	0	0	0
[-5, 5]	[-1, 1]	0	0	0	0	0	0	0
[-5, 5]	[-5, 5]	0	0	0	0	0	0	0
[-5, 5]	[-10, 10]	0	0	0	0	0	0	0
[-5, 5]	[-20, 20]	0	0	0	0	100	0	16.7
[-10, 10]	[-1, 1]	0	0	0	14	100	0	19
[-10, 10]	[-5, 5]	0	0	0	24	100	0	20.7
[-10, 10]	[-10, 10]	0	0	0	26	100	0	21
[-10, 10]	[-20, 20]	0	0	0	60	100	0	26.7
[-20, 20]	[-1, 1]	0	0	0	90	100	0	31.7
[-20, 20]	[-5, 5]	0	0	0	94	100	0	32.3
[-20, 20]	[-10, 10]	0	0	0	96	100	0	32.7
[-20, 20]	[-20, 20]	0	0	0	98	100	100	49.7

Since FOA is a stochastic optimization algorithm, the solution found each time may not be the same; therefore, each function is repeated 100 times. If the final searching quality is within  $10^{-4}$  of the optimal value, the run is called a success run and its iteration number will be stored. Two indexes named “percentage of success (PS)” and “average valid iteration number (AVIN)” are defined as follows [13]:

$$PS = \frac{m}{100} \times 100\%, \quad (9)$$

$$AVIN = \frac{\sum_{i=1}^m n_i}{m} \times 100\%,$$

where  $m$  denotes the number of success runs among 100 runs,  $n_i$  denotes the number of iteration of the  $i$ th success run.

The parameters of FOA are maxgen = 300, sizepop = 50, LR  $\in [(-1, 1), (-5, 5), (-10, 10), (-20, 20)]$ , and FR  $\in [(-1, 1), (-5, 5), (-10, 10), (-20, 20)]$ .

Table 2 shows PS and AVIN of FOA when solving the six testing functions. From Table 2, the following can be seen that.

- (1) PS of  $f_{GP}$ ,  $f_{SH}$ , and  $f_{BR}$  is always equal to zero no matter what values LR and FR are. It is known that the extreme points of the three functions are nonzero,

so FOA cannot solve the problems when the extreme point is nonzero.

- (2) When the scopes of FR and LR become large, PS of  $f_{RA}$ ,  $f_{SP30}$ , and  $f_{SP50}$  increase. It is known that the extreme points of the three functions are zero points, so FOA can solve the optimization problems when the extreme point is zero, if FR and LR are large enough.
- (3) The Average PS is very small no matter what values LR and FR are, so it is concluded that FOA cannot solve complex optimization problems effectively.

2.3. *Analysis of FOA.* Through the analysis of (5), it can be found that NGMS has some disadvantages which limit the performance of FOA. The disadvantages are listed below.

- (1) FOA cannot solve the optimization problems when there exist negative numbers in the domain, because  $S_i > 0$  according to (5).
- (2) When the value of  $x$ -axis and  $y$ -axis is fixed,  $S_i$  in (5) does not follow uniform distribution. (Proof is described in Appendix B.)  
Since  $S_i$  does not follow uniform distribution, the candidate solution cannot be uniformly generated in the domain; That is to say, NGMS cannot allow

the search to be performed uniformly in the domain, therefore fruit fly swarm loses its ability to search for a global optimum solution. That is why FOA cannot solve complex optimization problems effectively.

- (3) In (5), when the values of  $x\_axis$  and  $y\_axis$  are large and the scope of FR is small, the change of  $rand(FR)$  has little impact on the value of  $S_i$ ; therefore it is easy for  $S_i$  to fall into local optimal point.
- (4) In (1)-(2), with the increase of the scopes of LR and FR, the probability that the absolute value of  $x\_axis$  and  $y\_axis$  becomes large increases, so  $S_i$  in (5) is easy to fall near zero point, and this can explain why FOA can solve the optimization problems when the extreme point is zero.

### 3. LGMS-FOA

**3.1. Introduction of LGMS-FOA.** In order to overcome the above disadvantages, NGMS is replaced with a new linear generation mechanism of candidate solution (abbreviated as LGMS), and a LGMS-based improved FOA (abbreviated as LGMS-FOA) is proposed. The steps of LGMS-FOA are listed below.

*Step 1* (parameters initialization). The main parameters of LGMS-FOA are the maximum iteration number ( $maxgen'$ ), the population size ( $sizepop'$ ), the searching coefficient ( $n$ ), the initial weight ( $w_0$ ), and the weight coefficient ( $\alpha$ ).

*Step 2.* Linear generation mechanism of candidate solution.

*Step 2.1.* Initial fruit fly swarm location:

$$x\_axis' = n * rand(\text{domain of definition}). \quad (10)$$

*Step 2.2.* Give the random direction and distance for food finding of an individual fruit fly:

$$\begin{aligned} x_i' &= x\_axis' + w * rand(\text{domain of definition}), \\ w &= w_0 * \alpha^{\text{gen}}. \end{aligned} \quad (11)$$

*Step 2.3.* Let the smell concentration judgment value ( $S_i'$ ) equal  $x_i'$ :

$$S_i' = x_i' = x\_axis' + w * rand(\text{domain of definition}). \quad (12)$$

*Remark 3.* Equation (12) is called LGMS.

*Step 3.* Calculate the smell concentration ( $Smell_i'$ ) of the individual fruit fly location by input the smell concentration judgment value ( $S_i'$ ) into the smell concentration judgment function (also called *objective function*):

$$Smell_i' = \text{objective function}(S_i'). \quad (13)$$

*Step 4.* Find out the fruit fly with maximal smell concentration among the fruit fly swarm:

$$[Smell\ best' \ best\ index'] = \max(Smell'). \quad (14)$$

*Step 5.* Keep the maximal concentration value and  $x'$  coordinate. Then, the fruit fly swarm flies towards that location by using vision:

$$\begin{aligned} Smell\ best' &= \text{best smell}', \\ x\_axis' &= x(\text{best index}'). \end{aligned} \quad (15)$$

*Step 6.* Enter iterative optimization to repeat the implementation of Steps 2–5. When the smell concentration is not superior to the previous iterative smell concentration any more or the iteration number reaches the maximal iteration number, the circulation stops.

The complete flowchart of LGMS-FOA is shown in Figure 3, which is listed in Appendix C.

**3.2. Advantage of LGMS-FOA.** Compared with NGMS, LGMS has some advantages which are listed below.

- (1) The range of  $S_i'$  in (12) can cover the whole scope of the domain.
- (2) When the value of  $x\_axis'$  is fixed,  $S_i'$  in (12) follows uniform distribution. So LGMS can allow the search to be performed uniformly in the domain; therefore fruit fly swarm enhances its ability to search for a global optimum solution.
- (3) In LGMS, a parameter called inertia weight is brought in to balance the global and local search. A large inertia weight facilitates a global search while a small inertia weight facilitates a local search. By decreasing the inertia weight from a large value to a small value, LGMS-FOA tends to have more global search ability at the beginning of the run while having more local search ability near the end of the run [8].

### 4. Numerical Simulation

The performances of LGMS-FOA and FOA are compared first, and then the performances of LGMS-FOA and other metaheuristics are compared.

#### 4.1. Comparison of LGMS-FOA with FOA

**4.1.1. Experimental Setup.** In order to compare the performances of FOA and LGMS-FOA, the same six functions in Table 1 are used, and every function is repeated 100 times.

The parameters of LGMS-FOA are  $maxgen' = 300$ ,  $sizepop' = 50$ ,  $n = 0.005$ ,  $w_0 = 1$ , and  $\alpha = 0.95$ .

The parameters of FOA are  $maxgen = 300$ ,  $sizepop = 50$ ,  $LR = [-20, 20]$ , and  $FR = [-20, 20]$ , because FOA with these parameters performs best in Table 2.

*Remark 4.* The function evaluation numbers of FOA and LGMS-FOA are the same in every iteration, because  $sizepop' = sizepop$ . This can ensure the fairness of comparison.

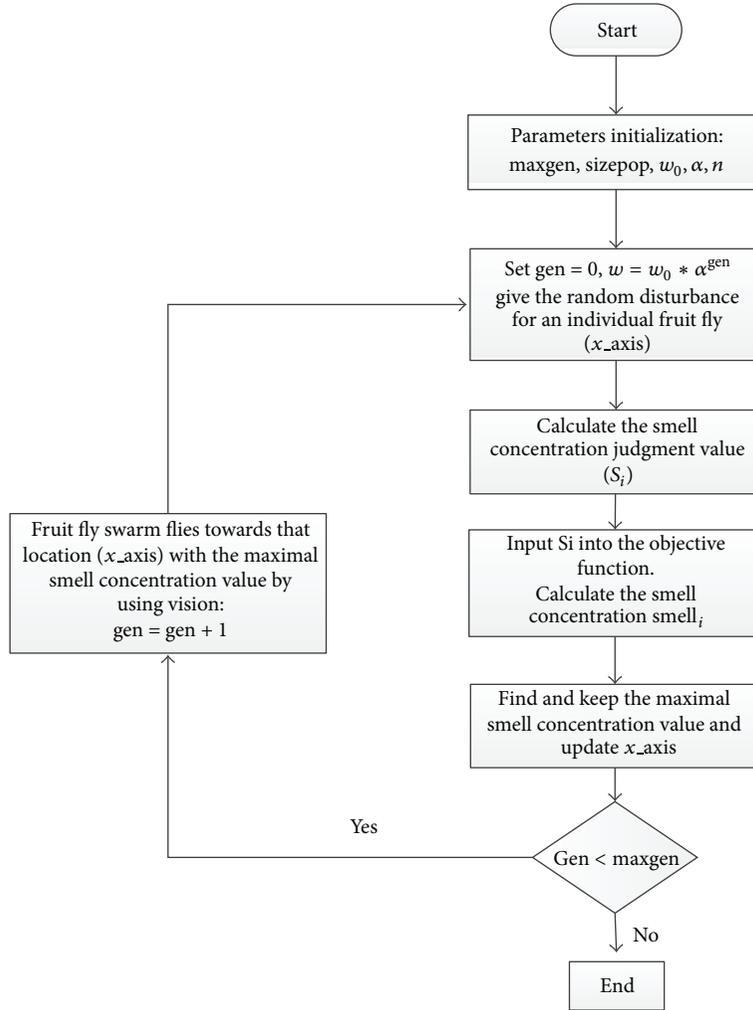


FIGURE 3: The flowchart of LGMS-FOA.

TABLE 3: Mean and standard deviation of LGMS-FOA and FOA.

$f$	Dimension	LGMS-FOA		FOA	
		Mean	Standard deviation	Mean	Standard deviation
$f_{GP}$	2	3.0000	$1.6851e - 09$	561.1129	134.2528
$f_{SH}$	2	-186.7309	$3.2454e - 11$	-37.8131	17.1154
$f_{BR}$	2	0.3979	$2.4248e - 14$	7.4194	4.8300
$f_{RA}$	2	-1.9879	0.0067	-1.9774	0.0182
$f_{SP30}$	30	$9.1722e - 09$	$1.7208e - 09$	$5.8495e - 06$	$1.5089e - 06$
$f_{SP50}$	50	$2.7285e - 04$	$1.810e - 3$	$4.0613e - 05$	$1.211e - 06$

4.1.2. *Experimental Results.* Table 3 shows mean and standard deviation of LGMS-FOA and FOA for 100 independent runs. Figures 4, 5, 6, 7, 8, and 9 show the performance of FOA and LGMS-FOA for solving the six testing functions which are listed in Appendix C.

From Table 3, it can be seen that mean of LGMS-FOA is much closer to the theoretical optima, and LGMS-FOA has better standard deviation than FOA when solving  $f_{GP}$ ,  $f_{SH}$ ,  $f_{BR}$ ,  $f_{RA}$ , and  $f_{SP30}$ . So it is concluded that LGMS-FOA is more effective and robust than FOA.

From Figures 4–9, it can be seen that the varying curves of objective values using LGMS-FOA descend much faster than those using FOA and the final searching quality of LGMS-FOA is better than FOA. So it is also concluded that LGMS-FOA is better than FOA.

4.1.3. *Robustness Analysis.* Table 4 shows PS and AVIN of FOA and LGMS-FOA when solving the six functions 100 times. From Table 4, it can be seen that LGMS-FOA can find global optima with very high PS for every function. Besides,

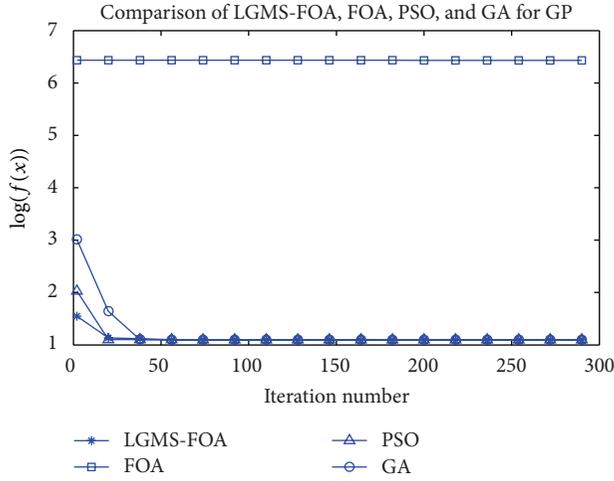
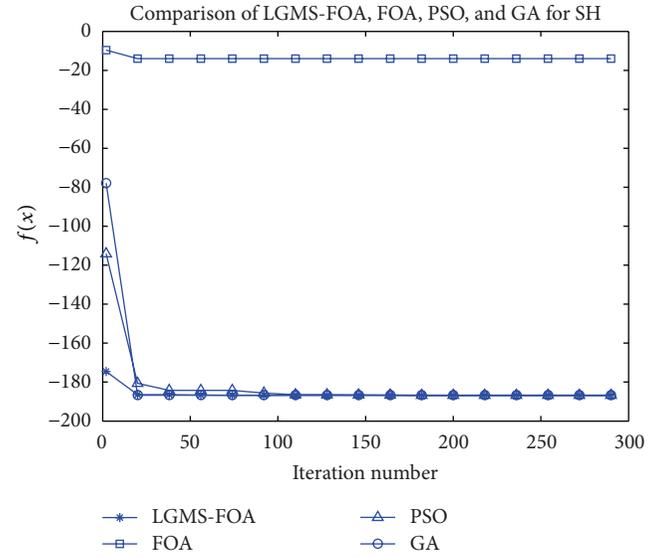
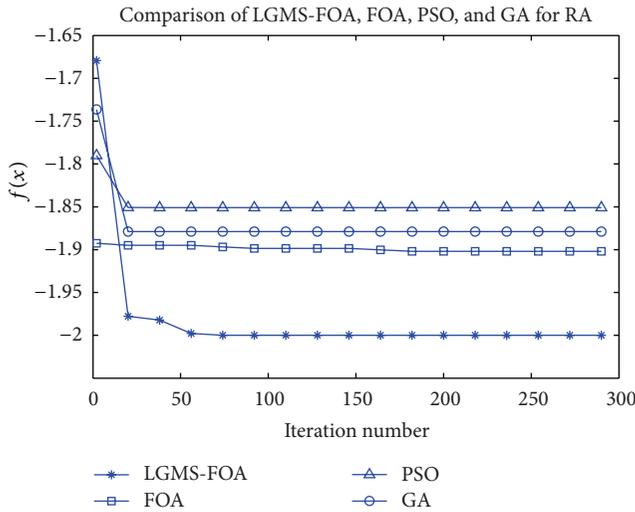
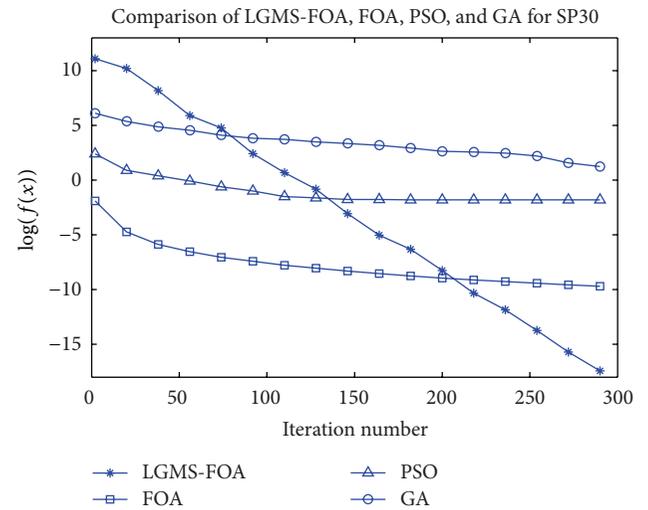
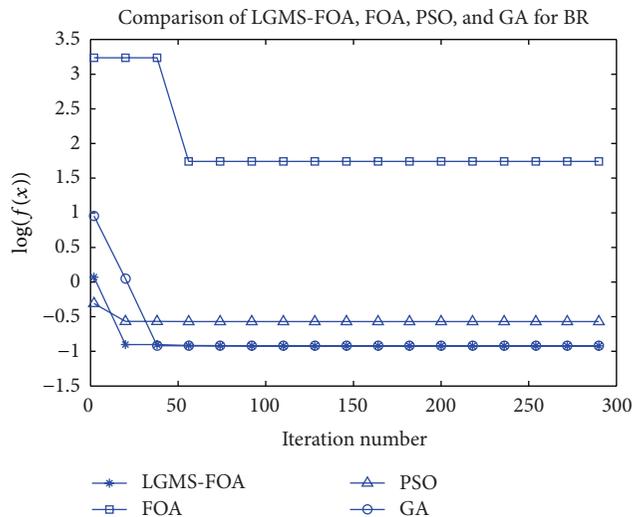
FIGURE 4: Comparison of LGMS-FOA, FOA, PSO, and GA for  $f_{GP}$ .FIGURE 7: Comparison of LGMS-FOA, FOA, PSO, and GA for  $f_{SH}$ .FIGURE 5: Comparison of LGMS-FOA, FOA, PSO, and GA for  $f_{RA}$ .FIGURE 8: Comparison of LGMS-FOA, FOA, PSO, and GA for  $f_{SP30}$ .FIGURE 6: Comparison of LGMS-FOA, FOA, PSO, and GA for  $f_{BR}$ .

TABLE 4: Robustness analysis.

$f$	Dimension	LGMS-FOA		FOA	
		PS (%)	AVIN	PS (%)	AVIN
$f_{GP}$	2	100	97	0	
$f_{SH}$	2	100	145	0	
$f_{BR}$	2	100	77	0	
$f_{RA}$	2	94	78	98	201
$f_{SP30}$	30	100	209	100	189
$f_{SP50}$	50	100	231	100	220
Average		99	140	49	203

for those valid runs, LGMS-FOA costs smaller AVIN than FOA. So, it is concluded that LGMS-FOA is more effective and reliable than FOA.

TABLE 5: Mean and standard deviation of GA and PSO.

$f$	Dimension	PSO		GA	
		Mean	Standard deviation	Mean	Standard deviation
$f_{GP}$	2	3.0000	$2.0311e - 15$	3.0000	$2.2430e - 15$
$f_{SH}$	2	-186.7309	$8.8118e - 13$	-184.2047	12.5013
$f_{BR}$	2	1.8374	2.0390	0.3979	$1.7252e - 12$
$f_{RA}$	2	-1.8169	0.1514	-1.9201	0.0797
$f_{SP30}$	30	0.2297	0.0986	2.6250	2.7102
$f_{SP50}$	50	1.3057	0.3687	3.5400	3.8184

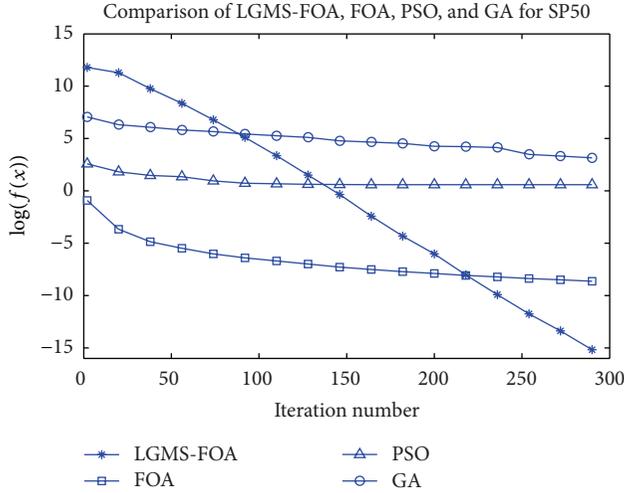


FIGURE 9: Comparison of LGMS-FOA, FOA, PSO, and GA for  $f_{SP50}$ .

4.2. Comparison of LGMS-FOA with Other Metaheuristics. In order to further show the effectiveness of LGMS-FOA, we carry out some comparisons with several other metaheuristics, such as the standard PSO [8] and the modified GA [2].

4.2.1. Experimental Setting. In PSO [8], the iteration number is 300, the population size is 50,  $c1 = 2$ ,  $c2 = 1.5$ ,  $w_{max} = 1.3$ ,  $w_{min} = 0.3$ , and  $v_{max}$  is limited to be 20% of the domain.

In GA [2], the iteration number is 300, the population is 50, Generation Gap (GGAP) is 0.8, stochastic universal sampling is used, the single dot cross operation with crossover probability is 0.7, and the discrete mutation with mutation probability is 0.1.

Testing functions are shown in Table 1, and every function is repeated 100 times.

Remark 5. The population sizes of GA, PSO, and LGMS-FOA are the same, so the function evaluation numbers of the three algorithms are also the same, and this can ensure the fairness of comparison.

4.2.2. Experimental Results and Discussion. Table 5 shows mean and standard deviation of GA and PSO of 100 independent runs. Table 6 shows PS of GA, PSO and LGMS-FOA of 100 independent runs. Figures 4–9 also show the performance of PSO and GA for solving the six testing functions.

TABLE 6: Robustness analysis.

$f$	Dimension	PS (%)		
		LGMS-FOA	PSO	GA
$f_{GP}$	2	100	100	100
$f_{SH}$	2	100	100	96
$f_{BR}$	2	100	1	100
$f_{RA}$	2	94	0	98
$f_{SP30}$	30	100	0	0
$f_{SP50}$	50	100	0	0
Average		99	33.5	65.6

From the comparison between Tables 4 and 5, it can be found that mean and standard deviation of LGSM-FOA is better than GA when solving  $f_{SH}$ ,  $f_{BR}$ ,  $f_{RA}$ ,  $f_{SP30}$ , and  $f_{SP50}$ . It can be also found that mean and standard deviation of LGSM-FOA is better than PSO when solving  $f_{BR}$ ,  $f_{RA}$ ,  $f_{SP30}$ , and  $f_{SP50}$ . So we can conclude that LGSM-FOA is more efficient than PSO and GA when the evaluation numbers of functions are the same.

From Figures 5, 6, 8, and 9, it can be seen that the final searching quality of LGMS-FOA is better than PSO and GA. From Figures 5 and 6, it can be seen that LGMS-FOA can greatly improve and speed up the convergence. So overall, the performance of LGMS-FOA is better than PSO and GA.

From Table 6, it can be found that LGMS-FOA can find global optima with higher PS than GA and PSO. So it is concluded that LGSM-FOA is more reliable.

## 5. Conclusion

This paper finds some disadvantages of FOA and proposes an improved FOA which is named LGMS-FOA. Simulations and comparisons of LGMS-FOA with FOA and other metaheuristics illustrate that LGMS-FOA is more effective and reliable. The future work is to apply LGMS-FOA for some real engineering optimization problems.

## Appendices

### A. Six Famous Nonlinear Functions Used in This Paper

(1)  $f_{GP}$ : Goldstein-Price, ( $n = 2$ ):

$$f = [1 + (x_1 + x_2 + 1)^2 (19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)]$$

$$\times \left[ 30 + (2x_1 - 3x_2)^2 (18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2) \right]. \quad (\text{A.1})$$

Global optimal solution: 3

Searching domain:  $-2 \leq x_i \leq 2, i = 1, 2.$

(2)  $f_{\text{BR}}$ : Branin, ( $n = 2$ ):

$$f = \left( x_2 - \frac{5.1}{4\pi^2} x_1^2 + \frac{5}{x_1} x_1 - 6 \right)^2 + 10 \left( 1 - \frac{1}{8\pi} \cos x_1 + 10 \right). \quad (\text{A.2})$$

Global optimal solution: 0.397887

Searching domain:  $-5 \leq x_1 \leq 10, 0 \leq x_2 \leq 15$

(3)  $f_{\text{RA}}$ : Rastrigin, ( $n = 2$ ):

$$f = x_1^2 + x_2^2 - \cos 18x_1 - \cos 18x_2 \quad (\text{A.3})$$

Global optimal solution:  $-2$

Searching domain:  $-1 \leq x_i \leq 1, i = 1, 2.$

(4)  $f_{\text{SH}}$ : Shuber, ( $n = 2$ ):

$$f = \left[ \sum_{i=1}^5 i \cos((i+1)x_1 + i) \right] \times \left[ \sum_{i=1}^5 i \cos((i+1)x_2 + i) \right]. \quad (\text{A.4})$$

Global optimal solution:  $-186.7309$

Searching domain:  $-10 \leq x_i \leq 10, i = 1, 2.$

(5)  $f_{\text{SP30}}$ : Sphere, ( $n = 30$ ):

$$f = \sum_{i=1}^{30} x_i^2. \quad (\text{A.5})$$

Global optimal solution: 0

Searching domain:  $-10 \leq x_i \leq 10, i = 1, 2 \dots 30$

(6)  $f_{\text{SP50}}$ : Sphere, ( $n = 50$ ):

$$f = \sum_{i=1}^{50} x_i^2. \quad (\text{A.6})$$

Global optimal solution: 0

Searching domain:  $-10 \leq x_i \leq 10, i = 1, 2 \dots 50.$

## B. Proof

**Proposition B.1.** If  $LR = [-1, 1], FR = [-5, 5],$

$x\_axis = a \in \text{rand}(LR), y\_axis = b \in \text{rand}(LR),$   
where  $a$  and  $b$  are constants;

$m \in \text{rand}(FR), n \in \text{rand}(FR),$  where  $m$  and  $n$  are variables;

$S_i = 1/((x+m)^2 + (y+n)^2),$  the domain of  $S_i$  is  $(0,10).$

Then,  $S_i$  does not follow uniform distribution.

*Proof.* For the sake of simplicity, we set  $a = 0$  and  $b = 0$  such that  $S_i = 1/(m^2 + n^2).$

(Proof by reductio ad absurdum.) Conversely, we assume that  $S_i$  is a random variable with uniform distribution on the interval  $(0,10);$  then  $P(S_i \leq 1) = 1/10.$

Since that  $m$  and  $n$  are independent and both uniformly distributed on  $[-5, 5],$  so the joint distribution of  $m$  and  $n$  is  $f(m, n) = f(m)f(n) = 1/100.$  Hence, we have

$$\begin{aligned} P(S_i \leq 1) &= P\left(\frac{1}{m^2 + n^2} \leq 1\right) = P(m^2 + n^2 \geq 1) \\ &= \int_{-1}^1 \int_{-\sqrt{1-m^2}}^{\sqrt{1-m^2}} f(m, n) dn dm \\ &\quad + \int_{-5}^{-1} \int_{-5}^5 f(m, n) dn dm \\ &\quad + \int_1^5 \int_{-5}^5 f(m, n) dn dm \\ &= \frac{\pi}{100} + 0.8 \neq 0.1. \end{aligned} \quad (\text{B.1})$$

This contradicts with  $P(S_i \leq 1) = 1/10.$  Therefore the previous assumption is not satisfied, and  $S_i$  does not follow uniform distribution.  $\square$

## C. Comparison of LGMS-FOA, FOA, PSO, and GA

See Figures 3–9.

## Acknowledgment

This paper is supported by the National Natural Science Foundation of China (Grant no. 71232004).

## References

- [1] Y. D. Bertrand and D. Braba, "Feature selection by a genetic algorithm application to seed discrimination by artificial vision," *Journal of the Science of Food and Agriculture*, vol. 76, pp. 77–86, 1998.
- [2] Y.-J. Lei, S.-W. Zhang, X.-W. Li, and C.-M. Zhou, *Matlab Genetic Algorithm Toolbox and Its Application*, Xidian University Publishing House, Xi'an, China, 2005.
- [3] M. B. Aryanezhad and M. Hemati, "A new genetic algorithm for solving nonconvex nonlinear programming problems," *Applied Mathematics and Computation*, vol. 199, no. 1, pp. 186–194, 2008.
- [4] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 200, pp. 671–680, 1983.
- [5] Y. Jin and J. Branke, "Evolutionary optimization in uncertain environments—a survey," *IEEE Transactions on Evolutionary Computation*, vol. 9, no. 3, pp. 303–317, 2005.
- [6] M. Dorigo and L. M. Gambardella, "Ant colony system: a cooperative learning approach to the traveling salesman problem," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 53–66, 1997.

- [7] Y. Fukuyama and H. Yoshida, "A particle swarm optimization for reactive power and voltage control in electric power systems," in *Proceedings of the Congress on Evolutionary Computation*, pp. 87–93, May 2001.
- [8] Y.-H. Shi and R. C. Eberhart, "Empirical study of particle swarm optimization," in *Proceedings of IEEE International Conference on Evolutionary Computation*, pp. 1945–1950, Washington, DC, USA, 1999.
- [9] J.-Y. Wu, "Solving unconstrained global optimization problems via hybrid swarm intelligence approaches," *Mathematical Problems in Engineering*, vol. 2013, Article ID 256180, 15 pages, 2013.
- [10] W.-T. Pan, "A new fruit fly optimization algorithm: taking the financial distress model as an example," *Knowledge-Based Systems*, vol. 26, no. 2, pp. 69–74, 2012.
- [11] H.-Z. Li, S. Guo, C.-J. Li, and J.-Q. Sun, "A hybrid annual power load forecasting model based on generalized regression neural network with fruit fly optimization algorithm," *Knowledge-Based Systems*, vol. 37, pp. 378–387, 2013.
- [12] H.-L. Shieh, C.-C. Kuo, and C.-M. Chiang, "Modified particle swarm optimization algorithm with simulated annealing behavior and its numerical verification," *Applied Mathematics and Computation*, vol. 218, no. 8, pp. 4365–4383, 2011.
- [13] B. Liu, L. Wang, Y.-H. Jin, F. Tang, and D.-X. Huang, "Improved particle swarm optimization combined with chaos," *Chaos, Solitons and Fractals*, vol. 25, no. 5, pp. 1261–1271, 2005.

## Research Article

# Improved SpikeProp for Using Particle Swarm Optimization

Falah Y. H. Ahmed, Siti Mariyam Shamsuddin, and Siti Zaiton Mohd Hashim

*Soft Computing Research Group, Faculty of Computer Science & Information Systems, Universiti Teknologi Malaysia, 81310 Skudai, Johor, Malaysia*

Correspondence should be addressed to Siti Mariyam Shamsuddin; mariyam@utm.my

Received 27 March 2013; Accepted 1 July 2013

Academic Editor: Praveen Agarwal

Copyright © 2013 Falah Y. H. Ahmed et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A spiking neurons network encodes information in the timing of individual spike times. A novel supervised learning rule for SpikeProp is derived to overcome the discontinuities introduced by the spiking thresholding. This algorithm is based on an error-backpropagation learning rule suited for supervised learning of spiking neurons that use exact spike time coding. The SpikeProp is able to demonstrate the spiking neurons that can perform complex nonlinear classification in fast temporal coding. This study proposes enhancements of SpikeProp learning algorithm for supervised training of spiking networks which can deal with complex patterns. The proposed methods include the SpikeProp particle swarm optimization (PSO) and angle driven dependency learning rate. These methods are presented to SpikeProp network for multilayer learning enhancement and weights optimization. Input and output patterns are encoded as spike trains of precisely timed spikes, and the network learns to transform the input trains into target output trains. With these enhancements, our proposed methods outperformed other conventional neural network architectures.

## 1. Introduction

For all three generations of neural networks, the output signals can be continuously altered by variation in synaptic weights (synaptic plasticity). Synaptic plasticity is the basis for learning in all ANNs. As long as there is nonvariant activation function, accurate classification based on certain vector input values can be implemented with the help of a BP learning algorithm like gradient descent [1, 2].

Spiking neural network (SNN) utilises individual spikes in time domain to communicate and to perform computation in a manner like what the real neurons actually do [3, 4]. This method of sending and receiving individual pulses is called pulse coding where information which is transmitted is carried by the pulse rate. Hence, this type of coding permits multiplexing of data [2].

For instance, analysis of visual input in humans requires less than 100 ms for facial recognition. Yet, facial recognition was performed by [5] by using SNN with a minimum of 10 synaptic steps on the retina at the temporal lobe, allowing nearly 10 ms for the neurons to process. Processing time is short, but it is sufficient to permit an averaging procedure

which is required by pulse coding [2, 5, 6]. In fact, pulse coding technique is preferred when speed of computation is the issue [5].

## 2. Page Background and Related Works

*2.1. Spiking Neural Networks (SNNs).* Neural networks which perform artificial information processing are built using processing units composed of linear or nonlinear processing elements (a sigmoid function is widely used) [6–9]. SNN had remained unexplored for many years because it was considered too complex and too difficult to analyze. Apart from that, biological cortical neurons have long time constants. Inhibition speed can be of the order of several milliseconds, while excitation speed can reach several hundreds of milliseconds. These dynamics can considerably constrain applications that need fine temporal processing [10, 11].

Little is known about how information is encoded in time for SNNs. Although it is known that neurons receive and emit spikes, whether neurons encode information using spike rate or precise spike time is still unclear [12]. For those

supporting the theory of spike rate coding, it is reasonable to approximate the average number of spikes in a neuron with continuous values and consequently process them with traditional processing units (sigmoid, for instance). Therefore, it is not necessary to perform simulations with spikes, as the computation with continuous values is simpler to implement and evaluate [13].

An important landmark study by Maass [14] has shown that SNN can be used as universal approximations of continuous functions. Maass proposed a three-layer SNN (consisting of the input layer, the generalization layer, and the selection layer) to perform unsupervised pattern analysis. Reference [15] applied spiking neural networks to several benchmark datasets (which include internet traffic data, EEG data, XOR problems, 3-bit parity problems, and iris dataset) and performed function approximation and supervised pattern recognition [16].

One of the ongoing issues in SNN research is how the networks can be trained. Much research has been done on biologically inspired local learning rules [13, 17], but these rules can only carry out supervised learning for which the networks cannot be trained to perform a given task. Classical neural network research became famous because of the error-backpropagation learning rule. Due to this, a neural network can be trained to solve a problem which is specified by a representative set of examples. Spiking neural networks use a learning rule called SpikeProp which operates on networks of spiking neurons and uses the exact spike time temporal coding [18]. This means that the exact spike time of input and output spikes encodes the input and output values.

*2.2. Learning in Networks of Spiking Neurons (SpikeProp).* Learning in the perceptron networks is usually performed by a gradient descent method [19] by using the backpropagation algorithm [20], which explicitly evaluates the gradient of an error function. The same approach has been employed in the *SpikeProp* gradient learning algorithm [18] which learns the desired firing times of the output neurons by adapting the weight parameters in the Spike Response Model SRM0 [21]. Several experiments have been carried out on *SpikeProp* to clarify several burning issues, for example, the role of the parameter initialization and negative weights [22]. The performance of the original algorithm can be improved by adding the momentum term [23]. *SpikeProp* can be further enhanced with additional learning rules for synaptic delays, thresholds, and time constants [24], which will normally result in faster convergence and smaller network sizes for the given learning tasks. An essential speedup was achieved by approximating the firing time function using the logistic sigmoid [25]. Implementation of *SpikeProp* algorithm on recurrent network architectures has shown promising results [26].

*SpikeProp* does not usually allow more than one spike per neuron, which makes it suitable only for “time-to-first-spike” coding scheme [5]. Its adaptation mechanism fails for the weights of neurons that do not emit spikes. These difficulties are due to the fact that spike creation or its removal due to weight updates is very discontinuous. *ASNA-Prop* has been proposed [24] to solve this problem by emulating the feed

forward networks of spiking neurons with the discrete-time analog sigmoid networks with local feedback, which is then used for deriving the gradient learning rule. It is possible to estimate the gradient by measuring the fluctuations in the error function in response to the dynamic neuron parameter perturbation [27].

SpikeProp adopts error backpropagation procedures which have been used widely in the training of analog neural networks to perform supervised learning [28]. SpikeProp does have weaknesses. The first weakness concerns sensitivity to parameter initialization values, which means that if the neuron is still inactive after initialization, the SpikeProp will not perform training for these weights which will not produce any spike. The second weakness is that SpikeProp is only suitable in cases where there is latency-based coding. The third weakness is that SpikeProp works only for SNNs where neurons spike only once in the simulation time. Finally, SpikeProp algorithm has been designed for training the weights only. To address these weaknesses, several improvements to SpikeProp algorithms have been suggested [29].

### 3. Methodology

We introduce new learning rules and enhancement architecture for improved SpikeProp.

*3.1. Enhancement of SpikeProp Architecture by PSO (PSO-SpikeProp) (Model 1).* In this proposed method, the SpikeProp was accelerated using four basic parameters of PSO; these parameters are the acceleration constant for  $g_{best}$ , the acceleration constants for  $p_{best}$ , the time interval ( $\Delta t$ ), depending on the time of SpikeProp, and the number of particles used in SpikeProp.

The acceleration constants are used in the simulation to specify swarm behavior of particles.  $p_{best}$  and  $g_{best}$  positions are formed by the constants  $c_1$  and  $c_2$ . The global best solution over the particle depends on the pulse time of SpikeProp (this is given by the constant  $c_1$ ). The individual personal best solution over the particle depends on the pulse time of the SpikeProp (this is given by the constant  $c_2$ ). If  $c_1$  is more than  $c_2$ , the swarm moves around the global best solution. On the other hand, when  $c_2$  is greater than  $c_1$ , the swarm moves around the individual personal best.

The time  $\Delta t$  of the parameters in SpikeProp defines the time in each pulse interval over each movement that occurs in the solution space in the pools processes. Reducing these parameters in SpikeProp leads to higher granularity movement within the solution space and a higher  $d$  time value of the highest pulse in SpikeProp on the node. The higher numbers of particles in the swarm or simulation in SpikeProp form the greater amount of the space that is covered in the problem; hence, the optimization will be fewer.

Depending on the node that has the higher pulse time in SpikeProp solution space, the parameters can be adapted to achieve better optimization. These basic parameters in PSO are used by SpikeProp. There are subparameters which depend on the dataset of the problem (like particle dimension, number of particles, and the stopping condition).

TABLE 1: PSO parameters used in SpikeProp.

Parameters	Values
$c_1 (g_{best})$	1.0
$c_2 (p_{best})$	1.0
$\Delta t$ (time interval)	0.1
Number of particles	20
Problem dimension	Based on dataset SpikeProp architecture
Stop condition	SpikeProp minimum error or maximum number of iterations
Range of particles	From $-1$ to $1$ $[-1, 1]$

The number of particles in SpikeProp using PSO significantly affects the execution time. There is a tradeoff between the size of the practical swarms and the execution time. PSO with a well-selected parameter set can perform well under all circumstances [30].

In this study, the parameters that have been used are summarized in Table 1, while particle position (weight and bias values) is initialized randomly with initial position velocity value set at 0.

Each particle position of the swarm is represented by a set of the weights for the current iteration. The dimension of the practical swarm determines the weight number of the network. In order to minimize the learning error, the particle should move within the weight space. Updating the weight of the network means changing the position in order to reduce the number of iterations. For each iteration, a new velocity calculation takes place to determine the new particle position movement. A set of new weights is used to obtain the new error, thus a new position. For PSO, the new weights are registered even if there is no noticeable improvement. This process applies for all particles. The global best particle position is the one with the least number of errors. The training process stops when the target minimum error is reached or the numbers of computational processes exceed the number of iterations allowable. When the training is complete, the weights are used to compute the classification error for the training patterns. The same patterns are used to test the network by using the same set of weights.

No researcher has yet used SpikeProp on PSO.  $p_{best}$  value and  $g_{best}$  value are applied to solve problems associated with the learning error. The SpikeProp weight and SpikeProp bias are achieved by adding the calculated velocity value as shown in (1) and (2). A new set of positions is used to produce the new learning error. In this proposed method, the classification dataset output has been written in a minimum number of iterations with the lowest error. The summary on PSO-SpikeProp learning process is shown in Figure 1.

$$w_{ij,new} = w_{ij,old} + \Delta w_{ij} * \Delta t \Big|_{t=t_j^d}^{(t)}, \quad (1)$$

$$\Delta w_{ij}(n) = c_1 * (p_{best,n} - w_{ij}(n)) + c_2 * (g_{best,n} - w_{ij}(n)) \Big|_{t=t_j^d}^{(t)}. \quad (2)$$

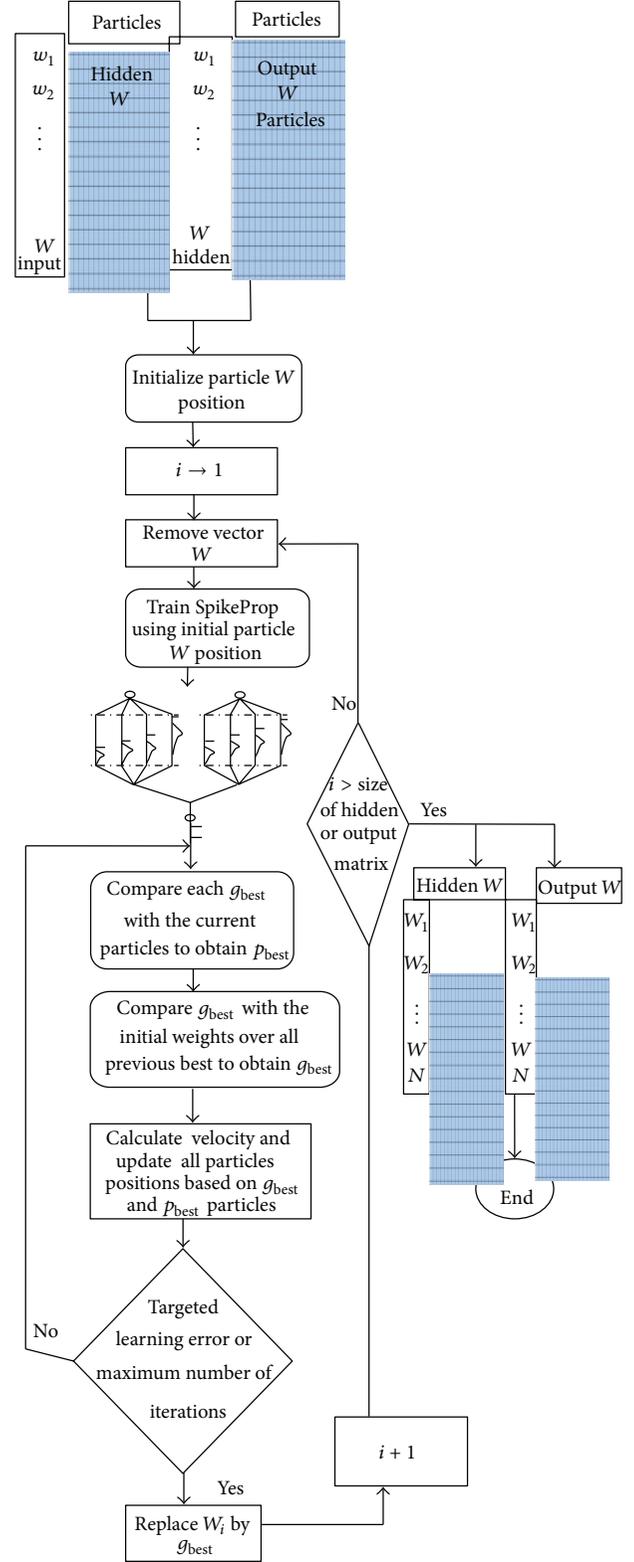


FIGURE 1: PSO-SpikeProp learning process.

In this proposed technique, the PSO is applied to SpikeProp algorithm (refer to Figure 1). Equation (1), the new weight ( $w_{ij,new}$ ), is performed for each element of the positions

of  $p_{\text{best}}$  and  $g_{\text{best}}$ . In (2),  $\Delta w_{ij}(n)$  subtracts the dimensional weight of the element from the dimension from the best vector and then multiplies this by a random number (between 0.0 and 1.0) and an acceleration constant ( $c_1$  and  $c_2$ ). A number of particles have been used to solve 8 different dataset problems. The objectives of the study are to reduce errors, enhance the learning rate of SpikeProp, and to speed up the algorithm process. Figure 1 shows that particle swarms with initial random rates have different mean squared error (MSE). During the learning process, all particles move together to get  $p_{\text{best}}$  and  $g_{\text{best}}$ .  $g_{\text{best}}$  fit is an optimum solution.

**3.2. Proposed  $\mu$  Angle Driven Dependency Learning Rate (Model 2).** The proposed  $\mu$  angle driven dependency learning rate is an extension of Chan's [31] adapted learning rate and momentum during the training used in BP. This proposed method enhances SpikeProp learning according to the angle calculation between  $\Delta E(n)$  and  $\Delta w(n-1)$ . The adaptation adjusts the angle at 90 degrees as per the Pythagoras formula method to get the square of the hypotenuse that equals to the sum of the squares of the other two sides. If the angle is less than 90 degrees, the learning rate is increased inversely, but if the angle is larger than 90 degrees, the learning rate is decreased. These mathematical methods have been applied to enhance SpikeProp, as shown next.

- (1) The angle  $\theta$  between the change of errors and the change of the weights can be calculated by

$$\cos \theta(n) = \frac{\Delta W(n-1)}{\sqrt{\Delta E(n)^2 + \Delta W(n-1)^2}} \Bigg|_{t=t_j^d}^{(t)}. \quad (3)$$

- (2) The adaption learning rate can be calculated by using the flowing formula:

$$\mu(n) = \mu(n-1) * (1 + 0.5 * \cos \theta(n)). \quad (4)$$

- (3) Adaption of the momentum can be acquired as follows:

$$\alpha(n) = \alpha(0) * \frac{\|\Delta E(n)\|}{\|\Delta W(n-1)\|} \Bigg|_{t=t_j^d}^{(t)}. \quad (5)$$

- (4) As mentioned previously, the weights can be changed as follows:

$$\Delta W_{ij}(n) = \mu(n) * \left( \frac{\partial E}{\partial W_{ij}} \right) + \alpha(n) * \Delta W_{ij}(n-1) \Bigg|_{t=t_j^d}^t. \quad (6)$$

Fortunately, the learning rate is adapted much faster when we are using the modified adaptation rule:

$$\mu(n) = \mu(n-1) * (1 + 0.1 * \cos \theta(n)). \quad (7)$$

Moreover, a backtracking strategy has been used, which reruns learning steps taking more than half learning rate time if total error increases. From this it can be concluded that the learning rate gets improved in Spikeprop with a higher rate than the standard SpikeProp.

**3.3. Merging Model 1 with Model 2 for Enhancing SpikeProp (Model 3).** In order to get better performance and enhance the operation of Spikeprop, a merging process between Model 1 and Model 2 (resulting in Model 3) has been carried out. Model 3 has an architecture which is partly PSO and partly angle driven dependency learning rate system. The flowchart in Figure 2 shows the general working of Model 3.

**3.4. Error Measurement Functions.** The target of the SpikeProp algorithm is to learn a set of target firing times, denoted by  $\{t_j^d\}$ , at the output neurons  $j \in J$  for a given set of input patterns  $\{P[t_j \cdots t_h]\}$ , where  $P[t_j \cdots t_h]$  defines a single input pattern described by single spike times for each neuron  $h \in H$ . Given the desired spike times  $\{t_j^d\}$  and actual firing times  $\{t_j^a\}$ , this error function is defined by

$$(a) \quad \text{MSE} = \frac{1}{n} \sum_{j=2}^n (t_{j2}^a - t_{j2}^d)^2. \quad (8)$$

This thesis uses other error functions such as RMSE (9), MAPE (10) and MAD (11) to get more validation to evaluate the accuracy of SpikeProp (SNN) that has been enhanced and proposed (Models 1, 2, and 3).

Consider

$$(b) \quad \text{RMSE} = \sqrt{\frac{1}{n} \sum_{j=2}^n (t_{j2}^a - t_{j2}^d)^2}, \quad (9)$$

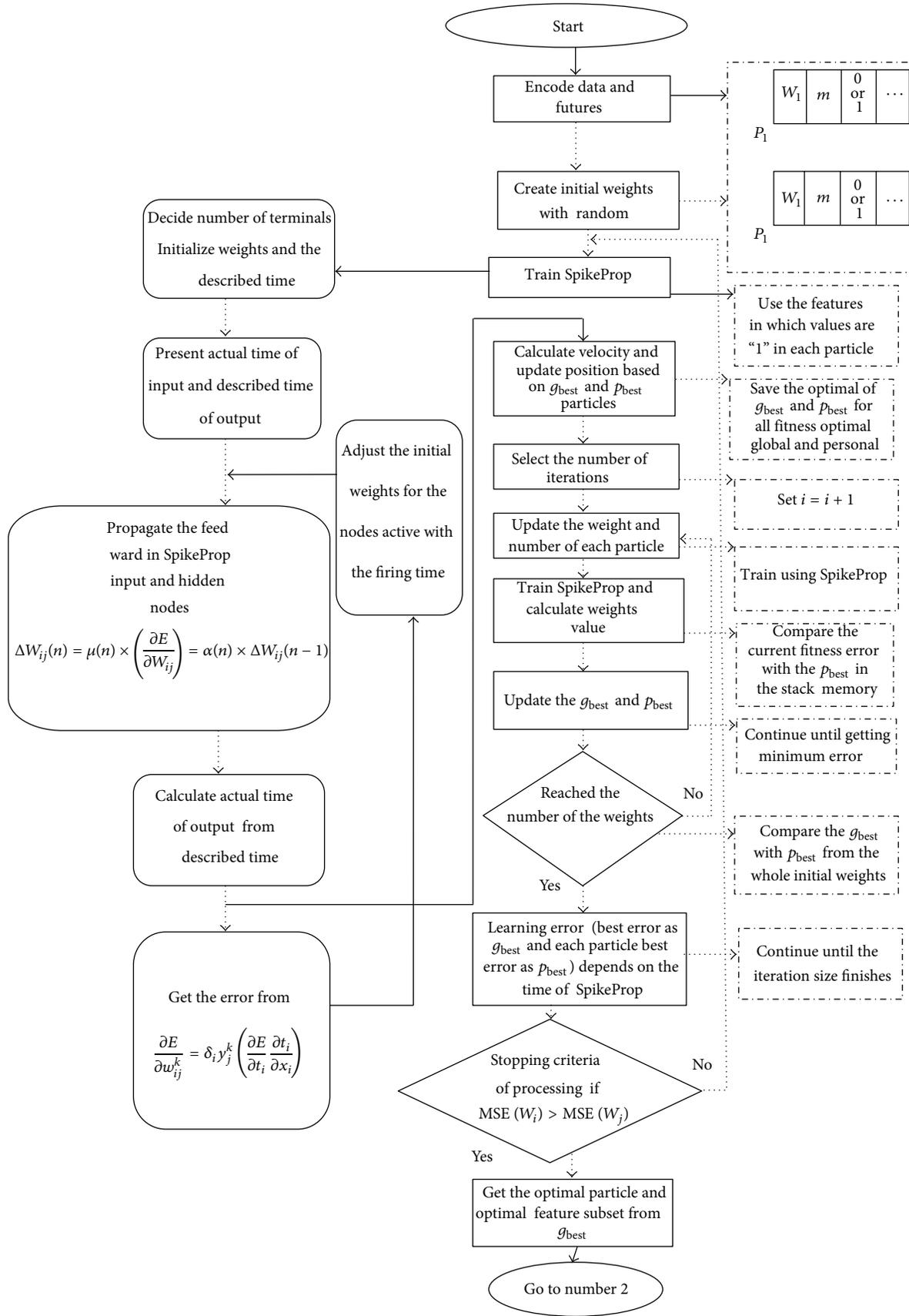
$$(c) \quad \text{MAPE} = \sum_{j=2}^n \left| \frac{t_{j2}^a - t_{j2}^d}{t_{j2}^d} \right| * \frac{100}{n}, \quad (10)$$

$$(d) \quad \text{MAD} = \sum_{j=2}^n \frac{|t_{j2}^a - t_{j2}^d|}{n}. \quad (11)$$

## 4. Results and Discussion

This section presents the results of study on learning of SpikeProp network based on the proposed method of improved SpikeProp. The results for all datasets involved are analyzed based on the convergence to MSE, RMSE, MAPE, and MAD with their classification performance. The results of the proposed methods for each dataset are analyzed based on performance (accuracy). For analysis purposes, methods of improved SpikeProp are used to train and optimize the networks, comparing different measurements of error. The results of SpikeProp based on the proposed method of improved SpikeProp are presented in the following subsections.

**4.1. Results and Analysis of Standard SpikeProp.** This section presents the result of standard SpikeProp for all datasets. The results are analyzed based on the convergence to MSE, RMSE, MAPE and MAD findings with their classification performance as shown in Table 2. All experiments for standard SpikeProp are based on ten runs. From Table 2, RMSE is better than MSE for the training datasets of BTX, Wine,



(a)

FIGURE 2: Continued.

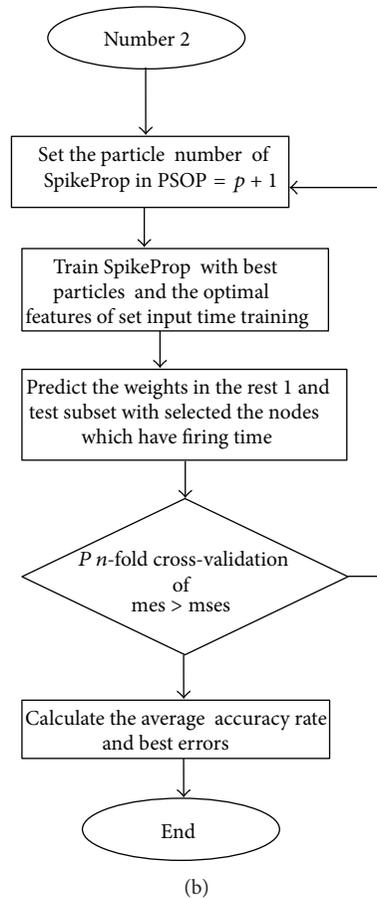


FIGURE 2: Merge Model 1 with Model 2 learning process.

Heart, Iris, Diabetes, Breast Cancer, Liver, Hepatitis, and XOR, respectively. The RMSE is also better than MAPE, and MAD error measurements for all datasets. These show that the SpikeProp algorithm also demonstrates in a direct way that networks of spiking neurons can carry out complex, nonlinear tasks in a temporal code. As the experiments indicate, the SpikeProp algorithm is able to perform correct classification on nonlinearly separable datasets with all types of errors measurement compared to traditional sigmoidal networks (BP) as shown in Table 3. Table 2 illustrates a comparison of the measurements between the training and testing of SpikeProp algorithm. From this table, we can see that the SpikeProp converges with less errors than standard BP.

**4.2. Results and Analysis of Standard BP.** The analyses of standard BP for all datasets are validated based on MSE, RMSE, MAPE, and MAD as shown in Table 3. From the training part in Table 3, the RMSE is better than MSE for the training datasets of Diabetes, Heart, Breast Cancer, Liver, Hepatitis, Wine, Iris, BTX, and XOR, respectively. The MSE is also better than MAPE and MAD for all datasets except BTX, which shows that MAD values are less compared to MSE. This traditional BP network is widely used by other researchers for

classification problems. Bohte et al. [18] designed SpikeProp for BP learning strategy; the comparisons between BP and SpikeProp are given in Tables 2 and 3. It shows that BP generates higher errors compared to SpikeProp.

**4.3. Analysis of the Proposed Model 1: PSO-Spikeprop.** Table 4 reveals the MSE generalization from the training part. MSE gives better results on Diabetes, Heart, Wine, and Breast Cancer datasets and least competitive for Liver, Hepatitis, Iris, BTX, and XOR datasets. For RMSE, the proposed PSO-SpikeProp is better for Wine, BTX, Diabetes, Iris, and Heart and less competitive in Liver, Breast Cancer, Hepatitis, and XOR, respectively. For MAD, the finding is better in Breast Cancer, Diabetes, and Heart and worse in Wine, Liver, Iris Hepatitis, BTX, and XOR, respectively. On the other hand, MAPE for PSO-SpikeProp gives higher error but still better than standard SpikeProp algorithm.

The PSO-SpikeProp gives the smallest error in MAPE compared to standard Spikeprop. However, PSO-SpikeProp stands out to be better if RMSE is being compared. Since the errors are squared before they are averaged, the RMSE gives a relatively low error rates, although the MSE has close values to RMSE as shown in Table 4. Therefore, Model 1: PSO-SpikeProp has its own good characteristics in generating

TABLE 2: Analysis for standard SpikeProp algorithm.

	Training SpikeProp				Testing SpikeProp			
	MSE	RMSE	MAPE	MAD	MSE	RMSE	MAPE	MAD
Breast cancer	0.549	0.496	43.305	0.992	0.571	0.511	44.419	1.023
BTX	1.713	0.225	36.294	1.575	2.190	0.269	44.451	1.888
Diabetes	0.426	0.386	31.556	0.773	0.492	0.420	34.348	0.840
Heart	0.442	0.374	32.058	0.749	0.367	0.335	29.820	0.671
Hepatitis	0.690	0.551	47.732	1.102	0.702	0.558	48.509	1.117
Iris	0.814	0.384	40.992	1.153	0.122	0.345	19.124	1.036
Liver	0.684	0.525	41.984	1.051	0.721	0.542	43.184	1.085
Wine	0.525	0.293	34.152	0.879	0.514	0.287	33.553	0.863

TABLE 3: Analysis for standard BP algorithm.

	Training BP				Testing BP			
	MSE	RMSE	MAPE	MAD	MSE	RMSE	MAPE	MAD
Breast cancer	0.672	0.525	54.759	1.278	0.816	0.638	54.443	1.277
BTX	1.982	0.796	88.018	2.575	2.707	0.798	88.344	2.588
Diabetes	0.510	0.429	37.576	1.271	0.792	0.627	59.208	1.255
Heart	0.608	0.477	41.902	1.270	0.805	0.633	55.724	1.267
Hepatitis	0.837	0.646	55.588	1.293	0.837	0.647	55.607	1.294
Iris	0.874	0.763	69.176	2.290	0.865	0.759	68.660	2.278
Liver	0.826	0.641	56.315	1.283	0.831	0.643	56.459	1.287
Wine	0.749	0.419	72.471	1.119	0.753	0.408	72.649	1.125

TABLE 4: Analysis for Model 1.

	Training Model 1				Testing Model 1			
	MSE	RMSE	MAPE	MAD	MSE	RMSE	MAPE	MAD
Breast cancer	0.356	0.361	35.261	0.723	0.367	0.372	36.268	0.745
BTX	1.270	0.199	35.596	1.398	1.763	0.247	43.785	1.730
Diabetes	0.241	0.245	22.650	0.490	0.284	0.277	25.778	0.555
Heart	0.2828	0.281	26.813	0.563	0.272	0.272	25.997	0.545
Hepatitis	0.483	0.448	42.640	0.896	0.494	0.457	43.609	0.915
Iris	0.351	0.251	33.270	0.754	0.112	0.147	18.38	0.443
Liver	0.360	0.340	31.854	0.680	0.383	0.357	33.332	0.714
Wine	0.312	0.196	25.033	0.590	0.305	0.196	25.269	0.589

the smallest error during the implementation. For this model we conclude that error can be reduced to reach the minimum value.

*4.4. Analysis of the Proposed Model 2: SpikeProp with Angle Driven Dependency ( $\mu$ ).* Similar to previous experiments, the results of Model 2 of SpikeProp with angle driven dependency are computed based on MSE, RMSE, MAPE, and MAD error measurements. Table 5 shows the findings of the model, which are based on ten independent runs on both training and testing datasets, respectively. The average testing errors are being calculated along with the standard deviations for all datasets.

As can be seen from Table 5, it is interesting to see the small standard deviations for all error rates on the training set

and the testing set in all datasets. The results of this proposed Model 2 have demonstrated that the generalization of MSE is better for Diabetes, Heart, and Wine datasets, while the results for Liver, BTX, Hepatitis, Iris, Breast Cancer, and XOR datasets are the least competitive. The results also have shown that the RMSE is better on all datasets and less competitive for other error measurements except for the Diabetes dataset which has better MSE values compared to RMSE. In general, the diversity of errors rates (MSE, RMSE, MAPE, and MAD) for this proposed Model 2 is considered better for all datasets. According to this model, we can find that the error is less than BP and SpikeProp standard.

*4.5. Analysis of the Proposed Model 3: Hybridization of Model 1 and Model 2.* Just as we get good strain by cross-breeding

TABLE 5: Analysis for Model 2.

	Training Model 2				Testing Model 2			
	MSE	RMSE	MAPE	MAD	MSE	RMSE	MAPE	MAD
Breast cancer	0.472	0.451	40.922	0.902	0.489	0.465	37.977	0.930
BTX	1.256	0.191	32.779	1.338	1.899	0.253	44.841	1.774
Diabetes	0.248	0.251	23.103	0.503	0.290	0.283	25.455	0.566
Heart	0.360	0.326	29.344	0.653	0.327	0.307	26.355	0.614
Hepatitis	0.523	0.500	42.233	1.001	0.452	0.435	43.929	0.872
Iris	0.438	0.278	34.668	0.835	0.117	0.377	18.335	1.131
Liver	0.419	0.379	34.285	0.759	0.444	0.397	34.289	0.795
Wine	0.365	0.229	28.329	0.688	0.358	0.229	28.679	0.689

TABLE 6: Analysis for Model 3.

	Training Model 3				Testing Model 3			
	MSE	RMSE	MAPE	MAD	MSE	RMSE	MAPE	MAD
Breast cancer	0.350	0.355	34.812	0.711	0.361	0.367	35.863	0.734
BTX	0.972	0.162	27.401	1.138	1.592	0.226	39.155	1.585
Diabetes	0.211	0.212	20.218	0.425	0.251	0.245	23.496	0.491
Heart	0.252	0.250	24.543	0.501	0.241	0.240	23.634	0.481
Hepatitis	0.437	0.421	41.179	0.843	0.446	0.431	42.194	0.863
Iris	0.330	0.244	32.747	0.732	0.105	0.153	17.576	0.461
Liver	0.359	0.339	31.778	0.678	0.380	0.356	33.296	0.712
Wine	0.263	0.168	22.186	0.506	0.273	0.178	23.637	0.535

two good genes, it may be possible to get good SpikeProp algorithm by merging (hybridizing) two good techniques. Therefore, this paper is concerned about the merging implementation of Model 1 and Model 2 (to get Model 3) as maintained in Figure 2.

In this section, we hybridize Model 1: PSO-SpikeProp and Model 2: SpikeProp with angle driven dependency to obtain a better performance for error rates (MSE, RMSE, MAPE, and MAD). We notice that the performance measurement is better than in the previous proposed method when the hybridization takes place as shown in Table 6. The experiments are based on 10 independent runs for the training and testing for all datasets, respectively. The results have revealed that generalization of error rates in RMSE is better for BTX, Wine, Diabetes, Iris, and Heart datasets and the least competitive for Breast Cancer, Hepatitis, Liver, and XOR datasets. Similarly, the result of MSE error rate has been demonstrated to be less better for Diabetes, Heart, Wine, Iris, Breast Cancer, Liver, Hepatitis, and BTX datasets, respectively. To get better performance, we hybridized Model 1 and Model 2, to get Model 3, to minimize the error to reach the optimum error value.

#### 4.6. Analysis and Result for Proposed Methods Based on Error.

In this section, the spiking neural network and SpikeProp use the encoding by depending on the timing of spike, where the first spike has a higher weight than the last one. Since a biological neuron uses few milliseconds to process information data, only few spikes are required and emitted, however the few first spikes with highest value information

can contribute to all process learning, as we used Gaussian function for encoding. Figures 3–10 show the convergence of the error and the number of iterations of the Spikeprop standard and the proposed methods for the improved SpikeProp in the classification Liver dataset, Breast Cancer, BTX, Diabetes, Heart, Hepatitis, Iris, and Wine data problems. The SpikeProp standard configuration had a much slower rate of convergence, as can be clearly seen in the plot. Although its rate of progress gradually slow down from the beginning till last iteration but kept in high error in the MSE, despite this slowdown in all data problems as shown in Figures 3–10.

*4.6.1. Analysis Error and Iterations of Model 1 (PSO-SpikeProp).* We also see from the convergence that the proposed method for improved SpikeProp is much better than standard SpikeProp as PSO-SpikeProp (Model 1) had dramatic slowdown from the first 10 iterations down to iteration number 40, and afterwards the plot got a slight slowdown in Liver as shown in Figure 3. In Breast Cancer data problem in Figure 4 the curve steps down at the start in error near to 0.5 until 0.38 exactly and then it continued descending to the last iteration at error 0.35 gradually. PSO-SpikeProp (model 1) is the first model from the proposed methods; the error turned down quickly in the first 10 iterations at 1.78 and continued to drop down till the last iteration in the error 1.27 in the BTX data problem as seen in Figure 5. Also in Diabetes data problem in Figure 6 the curve for the error steps down so fast in the first 10 iterations from 0.7 till 0.39 error and then continues to drop in a steady way until iteration number 86 of error 0.24. From the iteration number 87 till the last iteration

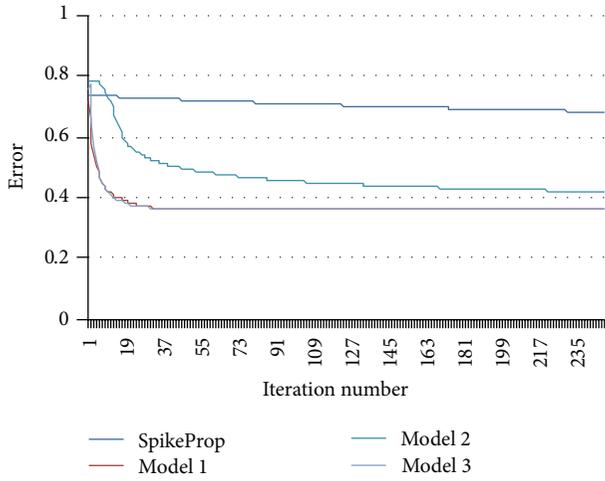


FIGURE 3: Liver.

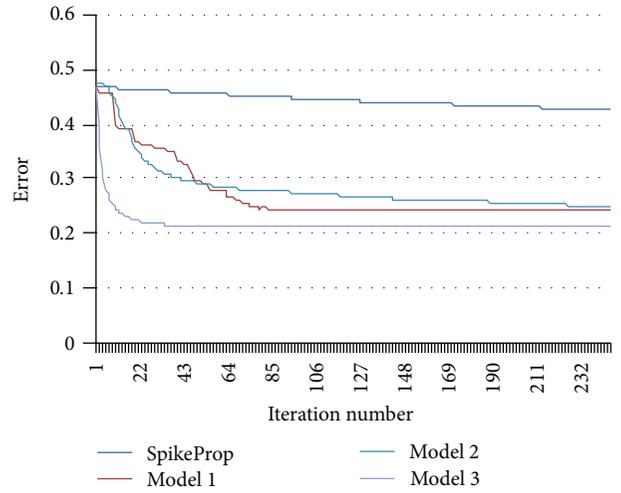


FIGURE 6: Diabetes.

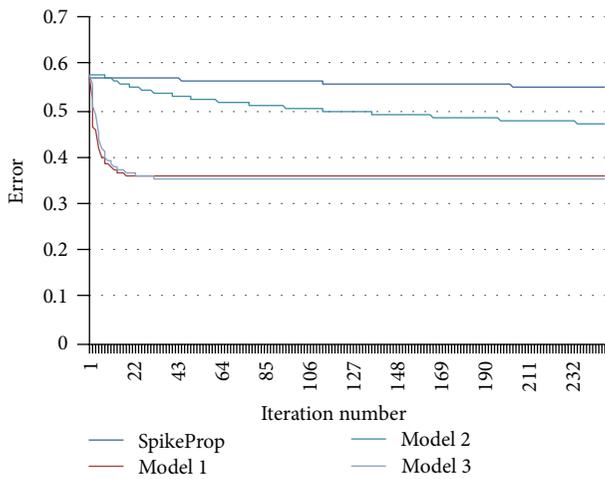


FIGURE 4: Breast cancer.

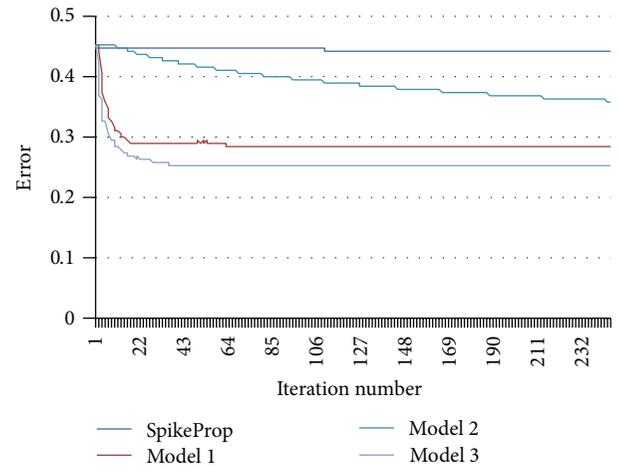


FIGURE 7: Heart.

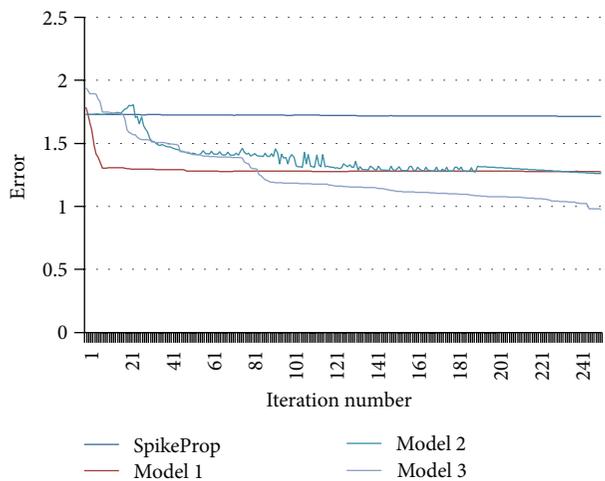


FIGURE 5: BTX.

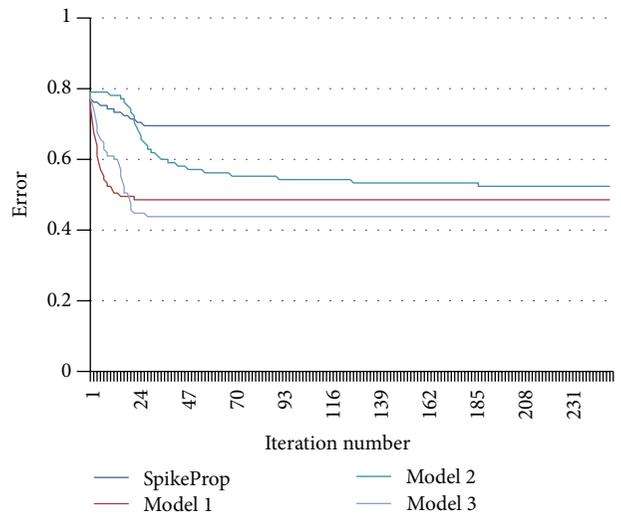


FIGURE 8: Hepatitis.

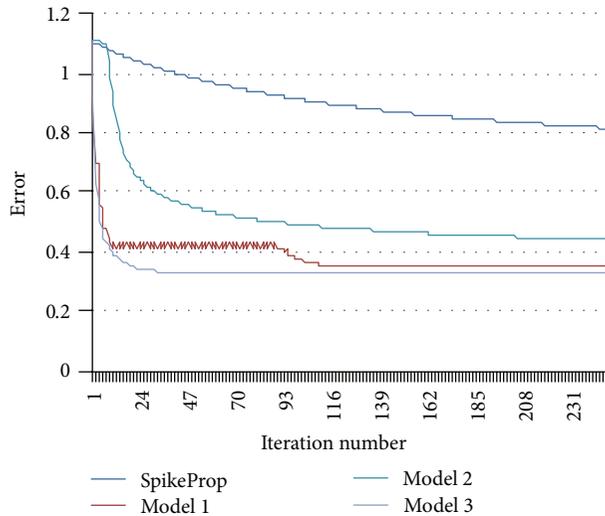


FIGURE 9: Iris.

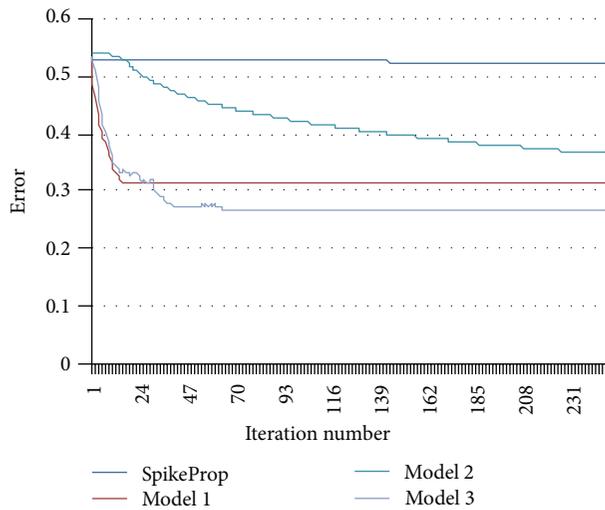


FIGURE 10: Wine.

it was a little sloped and stopped at the last iteration in error 0.24.

As we can see from Figure 7 in Heart problem the curve in error has dropped in a dramatic manner in the first 15 iterations in error from 0.45 to 0.28, and then the dropping is slowed down till iteration number 67 in error 0.28 and continued steadily till the last iteration for the same error. From the same proposed model in Hepatitis data problem in Figure 8 we can see that the slope of the curve starts at the first 15 iterations in a fast way in error 0.76 to 0.49, and then it slows down till iteration number 38 in error 0.48 and it stays likely stable till the last iteration. In Iris data problem in this model the plot from Figure 9 is stepped down in very fast drop from the first 10 iterations in error 1.105, and then it is zigzagged in error range 0.433 to 0.410 till iteration number 90 and then immediately fell down till the iteration number 112 in error 0.351 and it stayed on it till last iteration. Finally the plot of the error is stepped down in a high manner in the first

15 iterations of error 0.534 to 0.312, and then it stabilized in this error till the last iteration in Wine data problem as shown in Figure 10. From this model we have seen from Figures 3–10 how to obtain the list error from list iteration number from the whole data sets.

**4.6.2. Analysis Error and Iterations of Model 2.** In the second improvement for SpikeProp through the learning rate angle driven dependency (Model 2), we can also notice in Liver data problem as shown in Figure 3 that it has a slight dropping for the first 10 iterations in error around 0.75 and got a huge fall down till iteration number 50 for the error of nearly 0.55 and then it had a slight dropping till last iteration for the error 0.4. In the same model in Breast Cancer problem in Figure 4, the plot started to fall down on the first 20 iterations in error 0.57 until 0.47 gradually. Also the curve in BTX data problem as shown in Figure 5 start dropping down from the first 5 iterations in error 1.73 to 1.72, and it is fluctuated between the 6th iteration and iteration number 197 of the error range 1.73 to 1.31. Afterwards it has a steady drop till iteration 250 at error interval 1.30 to 1.25. From the result in Diabetes problem at the same Model 2 on the plot in Figure 6, the error in first seven iterations has dropped down slowly at error 0.47; after that it stepped down quickly till iteration number 20 in error 0.35 and then continued steadily until the last iteration for error 0.24. In the Heart data problem from Figure 7, the plot is stepped down from first iteration in error 0.45 until last iteration in error 0.36, and in some iterations, it is stable and then it continues descending. Also in Hepatitis data problem as shown in Figure 8, it has a stable error of 0.786 for the first 5 iterations, and then it start to fall down till iteration number 80 for error 0.549; after that it has a slower dropping till last iteration on error 0.523 in a sequence stepping down.

From Figure 9 as we can see clearly the curve is dropped down slowly in first 10 iterations in error 1.111 till 1.040; then it accelerates in dropping till iteration number 55 of error 0.53 and the drop slows down till last iteration in 0.438 error in the Iris data problem. Finally from Figure 10 in Wine data problem the plot shows Model 2 (learning rate angle driven dependency) is almost steady in first 15 iterations in error 0.538 to 0.526, and then the dropping got accelerated continuously till last iteration for error 0.366.

**4.6.3. Analysis Error and Iterations of Model 3 (SpikeProp with Angle Driven Dependency ( $\mu$ )).** (PSOSpikeProp and Learning Rate Angle Driven Dependency) or (model 3), in Liver dataset as it is clear from Figure 3 the plot it provides the best enhancement result for the convergence of error and number of iteration. The slant starts from the first iteration in the error a bit less than 0.8 till iteration number 20 impressively, and then a gradual descending has been done till last iteration of error close to 0.354. The curve from Figure 5 witnessed a dramatic drop in first 10 iterations starting from error 1.95 and the drop decreases till last iteration at error 0.97. We can notice from the curves in Figure 4 that the last model is much better from the SpikeProp standard and the previous methods of Breast Cancer dataset problem. The error steps down in a steady and fast manner from

TABLE 7: Result of training in terms of accuracy.

	SpikeProp	Training			
		Model 1	Model 2	Model 3	BP
Breast cancer	50.36	63.82	54.88	60.9	33.68
BTX	73.75	76.69	75.7	77.01	42.19
Diabetes	22.68	50.97	49.64	52.3	14.81
Heart	25.04	43.69	34.69	42.24	13.17
Hepatitis	8.21	21.38	17.57	22.23	2.44
Iris	57.68	72.27	68.22	73.22	32.79
Liver	4.86	31.97	24.01	29.3	2.14
Wine	56.03	70.47	67.57	70.36	35.78

TABLE 8: Result of testing in terms of accuracy.

	SpikeProp	Testing			
		Model 1	Model 2	Model 5	BP
Breast cancer	48.82	62.7	53.46	63.28	32.21
BTX	60.52	71.15	70.42	73.57	41.77
Diabetes	25.99	44.45	43.31	50.8	18
Heart	32.81	45.4	38.56	51.81	14.64
Hepatitis	8.78	18.42	16.5	23.65	4.33
Iris	77.66	84.66	80.73	96.03	50.03
Liver	8.55	28.55	20.48	28.79	2.8
Wine	56.84	70.53	67.53	73.24	40.13

the first iteration of error 1.93 to last iteration of error 0.97. From this we can see that most impact on SpikeProp standard is from PSO-SpikeProp and the leaning rate methods, and other proposed methods have less impact on BTX data problem as shown in Figure 5.

From the plot in Figure 6, this model started to drop down in a very fast way in first 10 iterations for error 0.7 to error 0.24, and then it started to slow down till iteration number 43 in error 0.21, and then it became almost stopped in error 0.21 till last iteration; we conclude from the comparison of the previous result that the model 3 gives the best and the least error from all other methods in Diabetes dataset. We can see from the plot that the curve fell down quickly in first 10 iterations for error range 0.45 to 0.29, and then the dropping started to slow down till iteration number 30 in error 0.258, then it got almost steady till last iteration in error 0.252. From the previous it is obvious that this model is the best among all the other models for Heart data problem as seen in Figure 7. This merge (Model 3) starts to drop down quickly from the first iteration for the error 0.787 until iteration number 30 for error 0.438, and then it stays stable until last iteration, as it is obvious from Figure 8 and the results that the third model has the most impact on SpikeProp compared to the previous models for Hepatitis data problem. Finally from Figure 9 it can be seen that the plot in this merge model started to step down quickly in first 30 iterations in error 1.107 to 0.332 and then it got steady for the same error till last iteration. This can show that the third model of improving SpikeProp gives the best result compared to other previous methods in Iris data problem.

Lastly, Model 3 is merging between Model 1 and Model 2 (PSO-SpikeProp and learning rate angle driven dependency); this merging model has a high impact on enhancing SpikeProp as it is seen in the curve of Figure 10 that in Wine data problem the slope is dropped quickly starting from first iteration of error 0.533 till iteration number 70 of error 0.263 and then it becomes almost stable till last iteration.

*4.7. Result and Analysis Comparison of the Proposed Methods in Terms of Accuracy.* This section displays the result of SpikeProp standard besides the proposed methods for enhanced SpikeProp measured in terms of accuracy. The experiments are run 10 s, 10 dependent runs on training and testing for all datasets, respectively (refer to Tables 7 and 8 and Figures 11 and 12). As it is shown in Table 7 for training, the first proposed method PSO-SpikeProp (Model 1) is evaluated in terms of accuracy; we can see that we got the value in Breast Cancer better than SpikeProp standard and proposed methods except Model 5. Regarding the BTX dataset problem, it is also better than SpikeProp standard and other proposed methods except Model 3. The generalization of accuracy for the proposed method PSOSpikeProp is better than SpikeProp standard and learning rate angle driven dependency (Model 2) in all datasets. Learning rate angle driven dependency is our proposed method; it is better than SpikeProp standard in all datasets. Finally, Model 3 is merging model (PSO-SpikeProp and Learning Rate Angle Driven Dependency) as illustrated in Figure 11 and Table 7 that it is better in accuracy generalization from all proposed methods and SpikeProp standard in all datasets.

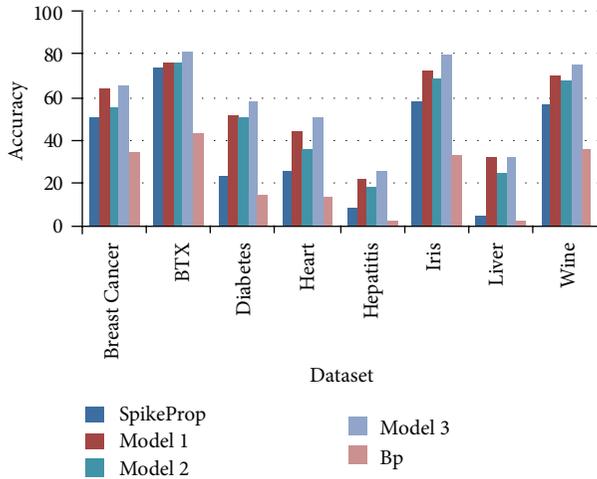


FIGURE 11: Results in training of the proposed methods in terms of accuracy.

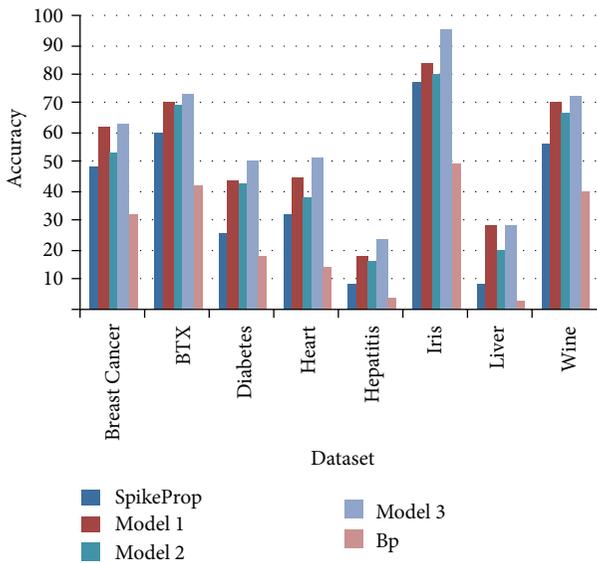


FIGURE 12: Results in testing of the proposed methods in terms of accuracy.

## 5. Conclusions

We introduced several extensions to the SpikeProp learning algorithm that make it possible to learn not only the weights, but also the delays and synaptic time constants of the connections and the thresholds of the neurons. Due to these enhancements, smaller network architecture can be used. This is mainly due to the fact that delays can now be trained and need not be enumerated. The simple 8 data sets could be solved with the same precision as the original SpikeProp algorithm, less errors (making the simulation and learning phase of the network much faster), and an increased learning convergence. There are several proposed models needed to improve the performance of SpikeProp further; hybridization of two or more good architectures is carried

out (for instance the hybridization of Model 1 and Model 2 to obtain Model 3). The purpose of hybridization is to leverage the best function from each component of the hybrid. As an example, Model 3 is the hybridization of Model 1 which is PSO-SpikeProp (enhancement Spikeprop architecture by PSO) with Model 2 which is SpikeProp enhancement using angle driven dependency learning rate. For Model 3, when the position of search is far from the optimum, PSO is used to directly move the point of search close to the optimum. When the search point is close to the optimum, Model 3 switches over to the system where there is SpikeProp enhancement using angle driven dependency learning rate to reach the optimum position. Also a thorough analysis of the weight initialization problem is required. The convergence rate seems to be pretty sensitive to this. Several techniques used in classic neural networks to speed up backpropagation learning could be added to SpikeProp to further speed up learning.

## References

- [1] Y. J. Jin, B. X. Shen, R. F. Ren, L. Yang, J. Sui, and J. G. Zhao, "Prediction of the styrene butadiene rubber performance by emulsion polymerization using backpropagation neural network," *Journal of Engineering*, vol. 2013, Article ID 515704, 6 pages, 2013.
- [2] W. Gerstner, R. Kempter, J. L. van Hemmen, and H. Wagner, *Hebbian Learning of Pulse Timing in the Barn Owl Auditory System in Maass*, 1999.
- [3] A. Belatreche and R. Paul, "Dynamic cluster formation using populations of spiking neurons," in *Proceedings of the 2012 International Joint Conference on Neural Networks (IJCNN '12)*, pp. 1–6, June 2012.
- [4] D. Ferster and N. Spruston, "Cracking the neuronal code," *Science*, vol. 270, no. 5237, pp. 756–757, 1995.
- [5] S. Thorpe, A. Delorme, and R. van Rullen, "Spike-based strategies for rapid processing," *Neural Networks*, vol. 14, no. 6-7, pp. 715–725, 2001.
- [6] N. Kasabov, "Evolving, probabilistic spiking neural networks and neurogenetic systems for spatio- and spectro-temporal data modelling and pattern recognition," in *Proceedings of the IEEE World Congress on Computational Intelligence (WCCI '12)*, vol. 7311 of *Lecture Notes in Computer Science*, pp. 234–260, 2012.
- [7] C. M. Bishop, *Neural Networks for Pattern Recognition*, 2000.
- [8] S. Haykin, *Neural Networks: A Comprehensive Foundation*, Prentice Hall PTR, 1998.
- [9] M. Negnevitsky, *Artificial Intelligence: A Guide to Intelligent Systems*, Addison Wesley, 2002.
- [10] N. Kasabov, "Integrative connectionist learning systems inspired by nature: current models, future trends and challenges," *Natural Computing*, vol. 8, no. 2, pp. 199–218, 2009.
- [11] Gewaltig, *Evolution of Synchronous Spike Volleys in Cortical Networks: Network Simulations and Continuous Probabilistic Models*, vol. 1822 of *Lecture Notes in Artificial Intelligence*, Springer, 2000.
- [12] S. Thorpe and J. Gastraits, "Rank order coding," in *Computational Neuroscience: Trends in Research*, pp. 113–118, Plenum Press, New York, NY, USA, 1998.
- [13] N. Kasabov, "To spike or not to spike: probabilistic spiking neuron model," *Neural Networks*, vol. 23, no. 1, pp. 16–19, 2010.

- [14] W. Maass, "Computing with spiking neurons," in *Pulsed Neural Networks*, W. Maass and C. Bishop, Eds., vol. 2, pp. 55–85, MIT Press, Cambridge, Mass, USA, 2001.
- [15] D. Mishra, A. Yadav, A. Dwivedi, and P. K. Kalra, "A neural network using single multiplicative spiking neuron for function approximation and classification," in *Proceedings of the International Joint Conference on Neural Networks (IJCNN '06)*, pp. 396–403, July 2006.
- [16] A. Grüning and I. Sporea, "Supervised learning of logical operations in layered spiking neural networks with spike train encoding," *Neural Processing Letters*, vol. 36, no. 2, pp. 117–134, 2012.
- [17] W. Gerstner and W. M. Kistler, *Spiking Neuron Models*, Cambridge University Press, 2002.
- [18] S. M. Bohte, J. N. Kok, and H. La Poutré, "Error-backpropagation in temporally encoded networks of spiking neurons," *Neurocomputing*, vol. 48, no. 1–4, pp. 17–37, 2002.
- [19] J. Šima, "Gradient learning in networks of smoothly spiking neurons," Tech. Rep. 1045, 2009.
- [20] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.
- [21] W. Gerstner and W. M. Kistler, *Spiking Neuron Models: An Introduction*, Cambridge University Press, 2002.
- [22] S. C. Moore, *Back-propagation in spiking neural networks [M.S. thesis]*, Department of Computer Science, University of Bath, Bath, UK, 2002.
- [23] J. Xin and M. J. Embrechts, "Supervised learning with spiking neural networks," in *Proceedings of the International Joint Conference on Neural Networks (IJCNN '01)*, pp. 1772–1777, July 2001.
- [24] B. Schrauwen and J. V. Campenhout, "Parallel hardware implementation of a broad class of spiking neurons using serial arithmetic," in *Proceedings of the 14th European Symposium on Artificial Neural Networks*, M. Verleysen, Ed., pp. 623–628, 2007.
- [25] K. Berkovec, *Learning in networks of spiking neurons [M.S. thesis]*, Faculty of Mathematics and Physics, Charles University, Prague, Czech Republic, 2006.
- [26] P. Tiño and A. J. S. Mills, "Learning beyond finite memory in recurrent networks of spiking neurons," *Neural Computation*, vol. 18, no. 3, pp. 591–613, 2006.
- [27] I. R. Fiete and H. S. Seung, "Gradient learning in spiking neural networks by dynamic perturbation of conductances," *Physical Review Letters*, vol. 97, no. 4, Article ID 048104, 2006.
- [28] K. Mergenthaler and R. Engbert, "Modeling the control of fixational eye movements with neurophysiological delays," *Physical Review Letters*, vol. 98, no. 13, Article ID 138104, 2007.
- [29] K. Nakazawa, M. C. Quirk, R. A. Chitwood et al., "Requirement for hippocampal CA3 NMDA receptors in associative memory recall," *Science*, vol. 297, no. 5579, pp. 211–218, 2002.
- [30] Y. Shi, *Particle Swarm Optimization*, IEEE neural Network Society, 2004.
- [31] L. W. Chan and F. Fallside, "An adaptive training algorithm for back propagation networks," *Computer Speech and Language*, vol. 2, no. 3–4, pp. 205–218, 1987.

## Research Article

# Nighttime Fire/Smoke Detection System Based on a Support Vector Machine

**Chao-Ching Ho**

*Department of Mechanical Engineering, National Yunlin University of Science and Technology, 123 University Road, Section 3, Douliou, Yunlin 64002, Taiwan*

Correspondence should be addressed to Chao-Ching Ho; hochao@yuntech.edu.tw

Received 22 May 2013; Accepted 14 August 2013

Academic Editor: Yudong Zhang

Copyright © 2013 Chao-Ching Ho. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Currently, video surveillance-based early fire smoke detection is crucial to the prevention of large fires and the protection of life and goods. To overcome the nighttime limitations of video smoke detection methods, a laser light can be projected into the monitored field of view, and the returning projected light section image can be analyzed to detect fire and/or smoke. If smoke appears within the monitoring zone created from the diffusion or scattering of light in the projected path, the camera sensor receives a corresponding signal. The successive processing steps of the proposed real-time algorithm use the spectral, diffusing, and scattering characteristics of the smoke-filled regions in the image sequences to register the position of possible smoke in a video. Characterization of smoke is carried out by a nonlinear classification method using a support vector machine, and this is applied to identify the potential fire/smoke location. Experimental results in a variety of nighttime conditions demonstrate that the proposed fire/smoke detection method can successfully and reliably detect fires by identifying the location of smoke.

## 1. Introduction

Video surveillance can be used to monitor fires and fire-related disasters. However, live surveillance systems require numerous people to monitor video screens continuously, leading to the possibility of human error; these systems also have other disadvantages including the need to store large amounts of data and the related high cost. Recently, more attention has been given to the detection of fire using surveillance cameras with machine vision which can detect smoke. Various researchers have proposed video fire/smoke detection methods based on signal processing and the related methods [1–8]. A video smoke detection technique comprising background subtraction, flickering extraction, contour initialization, and contour classification using both heuristic and empirical knowledge about smoke has been proposed [1]. The image processing approach involves the extraction of the smoke plume from the background using frame difference technologies. Image features such as motion, flickering, edge-blurring of moving segments, and edge-blurring of regions were extracted from the video to detect smoke. These features have been extracted using background

subtraction, temporal wavelet transformation, and spatial wavelet transformation. Smoke pixels were analyzed with a chromaticity-based static decision rule and a diffusion-based dynamic characteristic decision rule. A video smoke detection technique comprises background subtraction, flickering extraction, contour initialization, and contour classification using both heuristic and empirical knowledge related to the characteristics of smoke [2]. To avoid false flame detection which can result from interference caused by background illumination, neon colors, or traffic lights, the time-varying property of flame geometry is taken into account. When segmenting smoke features, color processing has advantages over grayscale processing. Color processing can avoid the generation of false alarms caused by variations in lighting conditions, for example, natural background illumination, better than grayscale processing can. Further, a video camera is a volume sensor, and it can potentially monitor a larger area. However, many current video-based fire surveillance systems have a major weakness regarding the detection of smoke at night. Video-based camera systems can only sense variations in the indoor environment when sufficient lighting is illuminating the monitored region. Illumination at night

is usually insufficient for smoke detection, and consequently smoke from a smoldering fire cannot be detected promptly.

To avoid the false flame detection caused by interference from background illumination and colored artificial light such as neon or traffic lights, the property of variation over time of flame geometry is taken into account [3]. In our previous work to detect fire flame and smoke using fuzzy logic and a continuously adaptive mean shift (CAMSHIFT) vision tracking algorithm was employed to provide feedback of the real-time position of fire flames and smoke using a high frame rate [2, 4]. To overcome the limitations related to illumination and the shortcomings of conventional smoke sensors, multisensor systems [5] and infrared cameras [6–8] have been used instead of regular visible-range cameras. Ruser and Magori [5] presented a fire detection method that used a combination of ultrasonic and microwave sensors to avoid false alarms. Liu et al. [6] presented a smoke recognition method based on active infrared charge-coupled device (CCD) video imagery. However, smoke detection using an active infrared camera to capture video at night is still difficult because of the high levels of background noise in the video. A false alarm for a fire may be caused by a variety of stimuli, such as a heater, sunlight, or a spotlight. Yuan et al. [7] proposed a fire detection method based on a plurality of infrared radiation arrays mounted in an area to detect smoke. Fang et al. [8] proposed the use of light section image detection to overcome the shortcomings of conventional beam-type smoke sensors designed to achieve early detection of fire/smoke. However, the drawback of these methods is that the detection system requires a setup involving many elements, such as infrared radiation arrays, infrared cameras, and signal processing units. Moreover, such systems are limited to monitoring only the flow of fire and/or smoke.

A fuzzy logic thresholding approach employed to provide the detection of nighttime fire/smoke in the earliest stages of a fire with a real-time alarm system was presented in our previous study [9]. However, the detection of fire signals will cause an alarm if the amount of smoke increases to the point that incident light scattering reaches a predetermined threshold, thus decreasing the reliability of the method in complex environments.

To address these problems, a smoke recognition method based on a color camera with active laser imagery that is able to sense smoke in a nighttime indoor environment under zero lux illumination is proposed. This paper introduces and analyzes current fire/smoke detection methods in Section 2. Section 3 then gives some experimental results and discussion. Finally, Section 4 provides the conclusions.

## 2. Nighttime Fire/Smoke Detection Using a Support Vector Machine (SVM)

To distinguish and classify candidate regions as fire/smoke or aliases, an SVM is used as a more sophisticated classifier than the fuzzy logic thresholding approach used in our previous research studies [9]. Support vector machines are data-driven and nonlinear approaches used for pattern classification problems which do not incorporate problem

domain knowledge [10–14]. Support vector machines are supervised learning techniques, which are a relatively new class of learning machines. Furthermore, an SVM not only effectively learns complex relationships without heuristic feature parameters but is also suited to the use of a limited trained data set.

The main goal of SVM theory is to use kernel functions to map the training data into high-dimensional feature space. A hyperplane is then found in this feature space that maximizes the margin between categories. The constructed hyperplane can then be used as a basis for classifying vectors of unknown classification with the largest margin between classes. Given the training sample  $(x_1, \dots, x_N)$  that are input patterns and  $(d_1, \dots, d_N)$  where  $d_i \in (+1, -1)^N$  are the corresponding desired response. The equation of a decision surface in the form of a hyperplane that does the separation is

$$w^T x + b = 0, \quad (1)$$

where  $x$  is an input vector,  $w$  is a weight vector, and  $b$  is a bias. The general form of the discriminant function is

$$g(x) = w^T x + b \quad (2)$$

and gives an algebraic measurement of the distance from  $x$  to the optimal hyperplane. To map the input vector to a high-dimensional feature space, the radial basis function (RBF) kernel is chosen as follows:

$$k(x, x_i) = \exp(-\gamma \|x - x_i\|^2), \quad (3)$$

where  $x$  is the input feature vector and  $x_i$  are the training samples. The symbol  $\gamma$  is the parameter that controls the width of RBF. In the proposed method, the feature vectors chosen were diffused laser light signal and scattered laser light signal from the extracted fire smoke candidate regions. These two features are then used as the input to the classifier.

To realize the production of an effective and efficient nighttime fire/smoke detection system, the proposed system is composed of a line-strip laser and a color CCD camera. The line-stripe laser emits light in the monitoring region of the camera. A general digital color video camera is used to capture several laser-projected smoke sample image sequences with a pixel resolution of  $1,280 \times 720$ . The fire/smoke detection process consists of four steps. First, the input color image is transformed into a YCbCr color space. YCbCr is a color format commonly used in digital video surveillance systems requiring data compression, where Y is the luminance, and Cb and Cr contain blue and red chrominance components. The YCbCr color space is applied instead of other color spaces because it can separate luminance from chrominance information. Experimentation has shown that the Y component apparently outperforms the other components as the diffusing and scattering laser signal received by the camera.

Image capturing and processing are the two major challenges involved in the construction of a laser light intensity measuring system. In the proposed fire/smoke detection system, the video signal capturing process is conducted using a low-cost CCD camera with a resolution of  $1280 \times 720$  pixels. The captured synchronized video frames are then transmitted

via USB to a personal computer. Then, the diffusing light intensity of the projected line-stripe laser is obtained by the camera. Next, the scattering light signal from the small and invisible-to-the-human-eye particles of fire/smoke is obtained using a color CCD camera. In the final step, the statistical distribution of the diffusing and scattering probability density is predicted by the SVM algorithm to determine the potential of actual fire/smoke as the result of the analysis.

### 3. Results and Discussion

Most fires in buildings start with a strong emission of fire and/or smoke. The particles and the air turbulence produced yield partial obstructions in the projecting laser section image, so the fire/smoke can be sensed. Smoke visibility refers to the attenuation of light or opacity along the line of sight, and the saturation will decrease when opacity increases. By relating the reduction in laser light intensity directly to smoke obscuration, the obscuring coefficient of the diffused laser light is defined as  $O_S$ . The  $O_S$  is represented as average pixel intensity values from a region of interest (ROI) window. A decrease in the light intensity of the sensed line-strip laser indicates that there is smoke in the region that is obscuring light from the line-strip laser, and the resulting light gradually becomes invisible to the camera. Smoke between the laser light source and the projected wall surface will decrease the quantity of incident light on the sensor. Light signal obscuration detection will cause an alarm if smoke particles block part of a laser light beam transmitted to a camera so that the laser light intensity decreases.

When particles of smoke from a fire interact with the laser beam, the light is scattered to the camera, whose light intensity response is a direct count of the number of particles [15]. The laser light scattered by the particles can be imaged to allow detection. The amount of smoke can be characterized by the scattered signal intensity received by the camera. Detection of scattered light signals will cause an alarm if the amount of smoke increases to the point where incident light scattering increases. The principle behind the operation of the proposed scattering light signal detection is to transform the camera surveillance area into a light scattering detection zone by directing a line-strip laser beam across it (Figure 1). Hence, the light scattered by particles of smoke from a fire in the beam can be used to calculate the smoke density detected by the camera. In order to have robust and continuous scattering feature values, scattering signals are accumulated in every frame. The particle sensing coefficient of the accumulative scattered laser light is defined as  $P_S$ . The  $P_S$  is represented as average pixel intensity values with corresponding masks having all nonzero values.

To create the proposed nighttime fire/smoke detection system, a general digital color video camera (Microsoft LifeCam HD-5000) was used to capture several fire/smoke sample image sequences, along with a harmless low-power laser light emitter (LT-B65100-GLD, 45 mW). The computer used in our system is a 2.5 GHz Pentium E5200 PC with 2 GB of RAM running Windows 7. The laser source emits a spatially modulated line-strip laser, and the laser beam is projected on a wall surface 8 meters from the camera. The camera receives

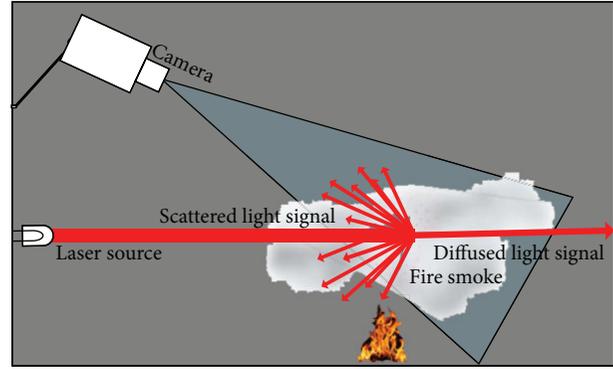


FIGURE 1: The proposed nighttime fire/smoke detection system.

TABLE 1: The fire/smoke test results by applying the proposed approach.

ID	Description	Event
1	Blank test	No alarm
2	Burning clothes	Fire alarm
3	Burning dried tree leaves	Fire alarm
4	Burning paper	Fire alarm
5	Burning woods	Fire alarm
6	Burning diesel oils	Fire alarm
7	Burning plastic bags	Fire alarm
8	Water mist spray to confuse	Nuisance

radiant energy from the diffusing and scattering line-strip laser beam and processes it to detect the existence of a fire and/or smoke conditions. Ten additional classified positive fire/smoke detections are also required prior to a decision to separate smoke and smoke-like aliases. The proposed system can readily be made to work in complete darkness. The laser and camera are mounted on a tripod at a distance of 7 m from the burning smoke box.

To verify that the measuring system effectively collects smoke information, a smoke sensor (i.e., HS-135) of a semiconductor type based on tin dioxide ( $\text{SnO}_2$ ) was employed [16].  $\text{SnO}_2$  sensors are widely used as a base material for commercial semiconductor gas sensors for detection of smoke. Smoke signals received by the HS-135 sensor are processed by the microcontroller, and then the microcontroller sends back numerical data to the computer via an RS232 serial link. The data acquired is recorded and synchronized with every video frame. The HS-135 sensor was installed above and near a smoke/burning test box at a height of 0.4 m above ground level. The fire/smoke was accurately classified by performing 10-fold cross-validation sets of training data (Tests 1–5).

The feasibility of the proposed approach was investigated with a series of eight test fires using different fuels. Figure 2 and Table 1 show the results of implementing the proposed fire smoke detection system under zero lux illumination (Tests 1–8). The zero lux intensity was confirmed using a digital lux meter (Testron Model TES-1332).

Figure 3 shows the results of evaluating the light obscuring and scattering characteristics of the tested images with



FIGURE 2: The case studies of the proposed fire/smoke detection system under zero lux illumination.

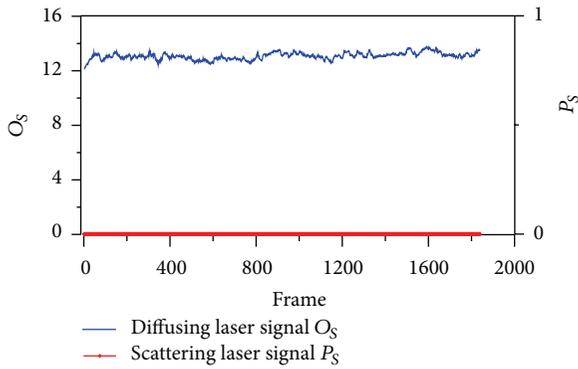


FIGURE 3: The first experiment: evaluate the results of the light obscuration and scattering characteristic features for the tested images with no fire/smoke inside the experimental room.

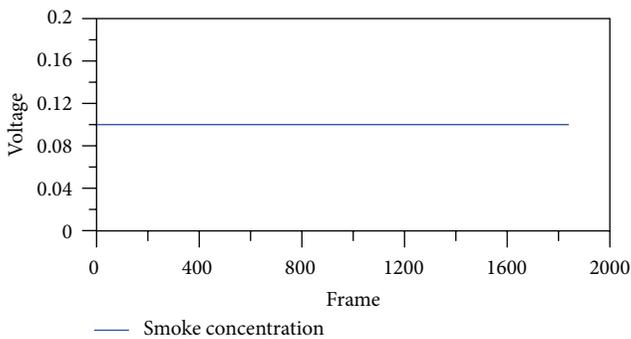


FIGURE 4: The first experiment: fire/smoke information is verified by the reading of smoke concentration from the HS-135 sensor output.

no fire/smoke inside the experimental room. Since the line-strip laser is not obscured, the value of diffused laser light coefficient  $O_S$  is close to a constant value. The employed off-the-shelf commercial laser diodes operate at a single wavelength. However, the wavelength is unstable because of fluctuations in the power supply and temperature [17]. Peak

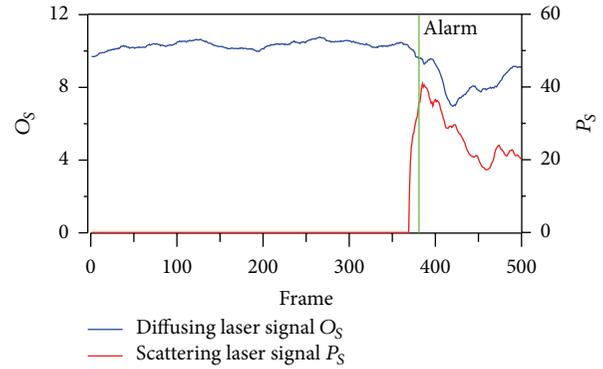


FIGURE 5: The second experiment: clothes were used as fuel.

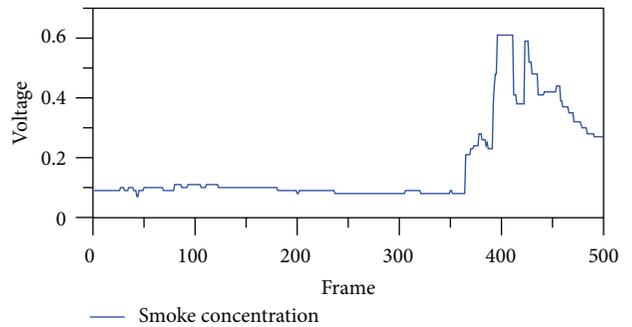


FIGURE 6: The second experiment: fire/smoke information is verified by the reading of smoke concentration from the HS-135 sensor output.

intensities vary as the wavelength varies and consequently the consecutive  $O_S$  value fluctuates (Figure 3). Also, no scattered laser is sensed, and so the value of the particle sensing coefficient  $P_S$  is zero. Fire/smoke information is verified by the reading of smoke concentration from the HS-135 sensor output (Figure 4).

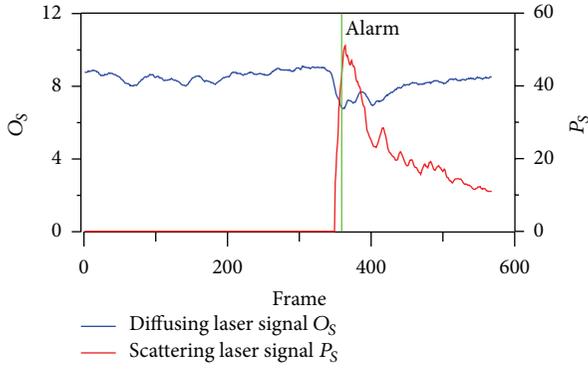


FIGURE 7: The experiment ID 3: dried tree leaves are selected as the burning material.

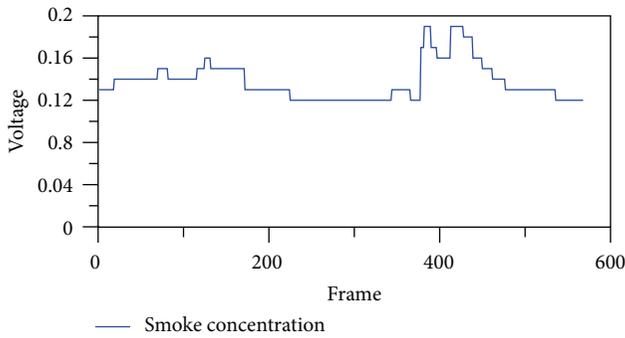


FIGURE 8: The experiment ID 3: fire/smoke information is verified by the reading of smoke concentration from the HS-135 sensor output.

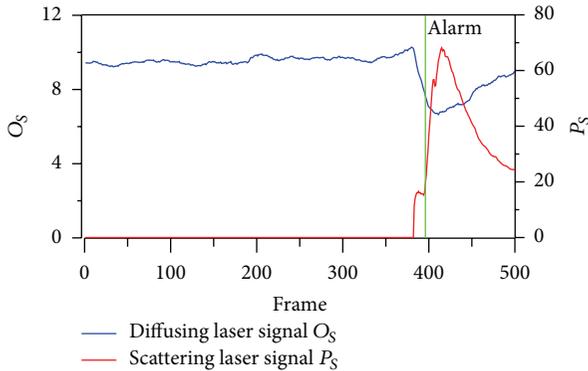


FIGURE 9: The experiment ID 4: papers are selected as the burning material.

In the second experiment, clothes are used as the burning material (Figure 5); the line-strip laser is obscured by smoke, and so the value of diffused laser light coefficient  $O_S$  decreases. Also, particles of smoke in the view region are scattering light from the laser beam, appearing as a visible blob to the camera; hence, the value of the particle sensing coefficient  $P_S$  increases. Since fire/smoke is turbulent and will fluctuate with periodic oscillations, the particle sensing coefficient  $P_S$  also fluctuates periodically. The potential fire/smoke was detected in the 381th frame with the summed-score

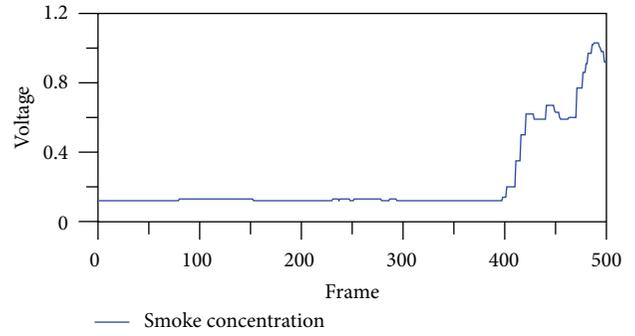


FIGURE 10: The experiment ID 4: fire/smoke information is verified by the reading of smoke concentration from the HS-135 sensor output.

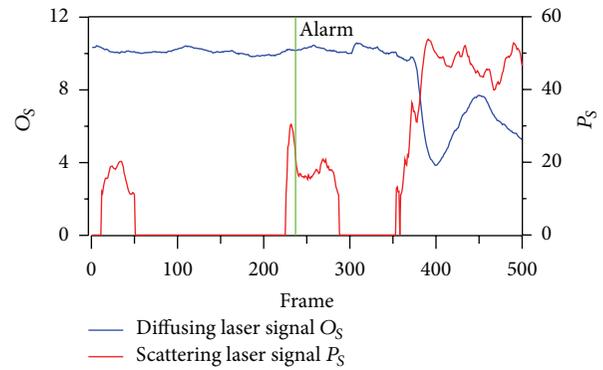


FIGURE 11: The experiment ID 5: woods are selected as the burning material.

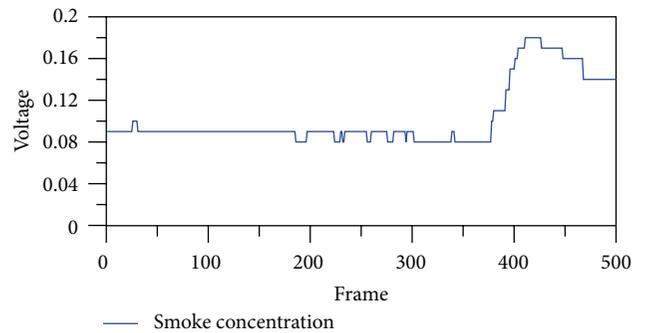


FIGURE 12: The experiment ID 5: fire/smoke information is verified by the reading of smoke concentration from the HS-135 sensor output.

level crossing the threshold, and the smoke concentration is verified by the HS-135 sensor output (Figure 6).

Experiments 3, 4, 5, 6, and 7 were conducted with various burning materials (Figures 7, 8, 9, 10, 11, 12, 13, and 14); clearly, the fire/smoke can be correctly extracted and an appropriate alarm sounded. In experiments 6 and 7, the HS-135 sensor failed to sense the smoke produced by plastic bags and diesel oils. However, the proposed approach did successfully detect

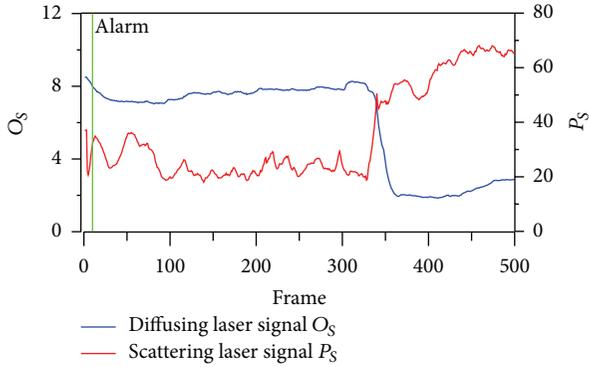


FIGURE 13: The experiment ID 6: diesel oils are selected as the burning material.

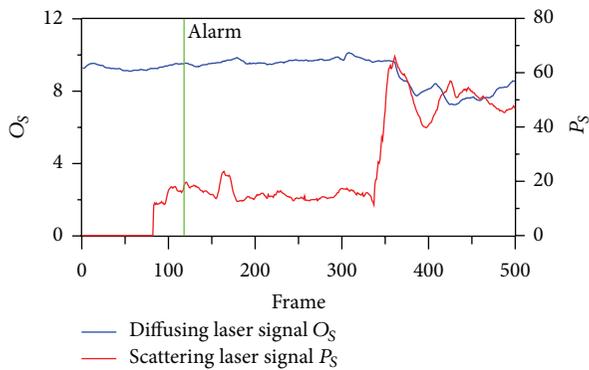


FIGURE 14: The experiment ID 7: plastic bags are selected as the burning material.

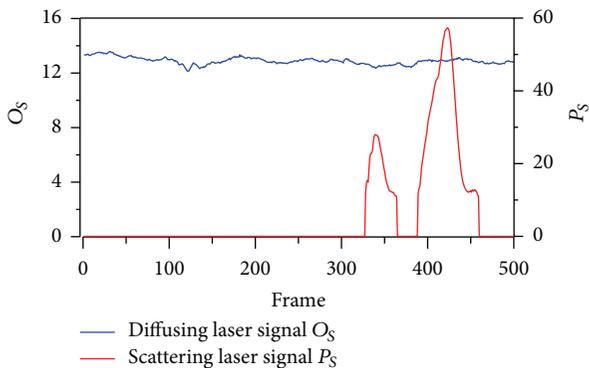


FIGURE 15: The experiment ID 8: water mist spray is selected for a false fire test.

the fire/smoke produced by diesel oils and plastic bags in the 10th frame and 118th frame (Figures 13, and 14).

In the last experiment, water mist spray was selected for a false fire alarm test with fire/smoke being produced in the burn box. The line-strip laser is not obscured by small particles, and hence the value of the diffused laser light coefficient  $O_S$  is close to a constant value. However, scattered laser light is sensed, and so the value of the particle sensing coefficient  $P_S$  is not zero (Figure 15). Then, the values of  $O_S$

and  $P_S$  are classified by the SVM system, and no alarm was sounded during Test 8 for a nuisance fire scenario.

## 4. Conclusions

This paper presents an approach which can be used to detect fire and/or smoke during the earliest stages of a fire with a real-time alarm system. Diffused laser light, scattered laser light signals, and an SVM classifier are employed to extract and detect real fire smoke images from the background image and are adopted to help validate the accuracy of the fire/smoke detection. The results show that fire and/or smoke can be successfully detected and the proposed algorithm can be integrated into existing surveillance systems to achieve the detection of fire/smoke recorded in video databases, as well as the real-time detection of fire. Future work is planned to fuse multisensors to improve the ability of the system to minimize false alarms and to accurately classify burning material.

## Acknowledgment

This research was partially supported by the National Science Council in Taiwan under Grant NSC 101-2221-E-224-008.

## References

- [1] Z. Xiong, R. Caballero, H. Wang, A. M. Finn, M. A. Lelic, and P. Y. Peng, "Video-based smoke detection: possibilities, techniques, and challenges," in *Proceedings of the Suppression, Detection and Signaling Research and Applications—aa Technical Working Conference (SUPDET '07)*, Orlando, Fla, USA, 2007.
- [2] C.-C. Ho, "Machine vision-based real-time early flame and smoke detection," *Measurement Science and Technology*, vol. 20, no. 4, Article ID 045502, 2009.
- [3] C. L. Lai, J. C. Yang, and Y. H. Chen, "A real time video processing based surveillance system for early fire and flood detection," in *Proceedings of the IEEE Instrumentation and Measurement Technology (IMTC '07)*, pp. 1–6, May 2007.
- [4] C.-C. Ho and T.-H. Kuo, "Real-time video-based fire smoke detection system," in *Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM '09)*, pp. 1845–1850, Singapore, July 2009.
- [5] H. Ruser and V. Magori, "Fire detection with a combined ultrasonic-microwave Doppler sensor," in *Proceedings of the International Ultrasonics Symposium*, pp. 489–492, Sendai, Japan, October 1998.
- [6] L. Y. Chang, Y. C. Yu, and Z. Y. Ming, "Nighttime video smoke detection based on active infrared video image," in *Proceedings of the International Conference on Electrical and Control Engineering (ICECE '10)*, pp. 1359–1362, Wuhan, China, June 2010.
- [7] H. Yuan, W. Fan, G. Su, B. Liu, S. Liu, and Q. Wang, "Method for detecting fire with light section image to sense smoke," United States Patent 6611207, 2003.
- [8] J. Fang, J. Jie, Y. Hong-Yong, and Z. Yong-Ming, "Early fire smoke movements and detection in high large volume spaces," *Building and Environment*, vol. 41, no. 11, pp. 1482–1493, 2006.
- [9] C. C. Ho and M. C. Chen, "Nighttime fire smoke detection system based on machine vision," *International Journal of*

- Precision Engineering and Manufacturing*, vol. 13, pp. 1369–1376, 2012.
- [10] V. Kecman, *Learning and Soft Computing: Support Vector Machines, Neural Networks, and Fuzzy Logic Models*, MIT Press, 2001.
- [11] L. Wang, *Support Vector Machines: Theory and Applications*, vol. 177, Springer, Berlin, Germany, 2005.
- [12] L. Wang and X. Fu, *Data Mining With Computational Intelligence*, Springer, Berlin, Germany, 2005.
- [13] A. Sloin and D. Burshtein, "Support vector machine training for improved hidden Markov modeling," *IEEE Transactions on Signal Processing*, vol. 56, no. 1, pp. 172–188, 2008.
- [14] A. H. Khandoker, M. Palaniswami, and C. K. Karmakar, "Support vector machines for automated recognition of obstructive sleep apnea syndrome from ECG recordings," *IEEE Transactions on Information Technology in Biomedicine*, vol. 13, no. 1, pp. 37–48, 2009.
- [15] D. W. Weinert, T. G. Cleary, G. W. Mulholland, and P. F. Beever, "Light scattering characteristics and size distribution of smoke and nuisance aerosols," in *Proceedings of the 7th International Symposium on Fire Safety Science*, pp. 209–220, Worcester, Mass, USA, June 2002.
- [16] H. Abderrahim, M. Berrebia, A. Hamou, H. Kherief, Y. Zanoun, and K. Zenata, "Measure of carbon dioxide using a gas sensor of a semiconductor type based on tin dioxide (SnO<sub>2</sub>)," *Journal of Materials and Environmental Science*, vol. 2, pp. 94–103, 2011.
- [17] S. Donati and L. Falzoni, "A PC-interfaced, compact laser-diode feedback interferometer for displacement measurements," *IEEE Transactions on Instrumentation and Measurement*, vol. 45, no. 6, pp. 942–947, 1996.

## Research Article

# Efficient Secure Multiparty Computation Protocol for Sequencing Problem over Insecure Channel

Yi Sun,<sup>1</sup> Qiaoyan Wen,<sup>1</sup> Yudong Zhang,<sup>2</sup> Hua Zhang,<sup>1</sup> and Zhengping Jin<sup>1</sup>

<sup>1</sup> State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China

<sup>2</sup> School of Computer Science and Technology, Nanjing Normal University, Nanjing, Jiangsu 210023, China

Correspondence should be addressed to Yi Sun; sybupt@bupt.edu.cn

Received 2 March 2013; Accepted 2 August 2013

Academic Editor: Vishal Bhatnagar

Copyright © 2013 Yi Sun et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

As a powerful tool in solving privacy preserving cooperative problems, secure multiparty computation is more and more popular in electronic bidding, anonymous voting, and online auction. Privacy preserving sequencing problem which is an essential link is regarded as the core issue in these applications. However, due to the difficulties of solving multiparty privacy preserving sequencing problem, related secure protocol is extremely rare. In order to break this deadlock, this paper first presents an efficient secure multiparty computation protocol for the general privacy-preserving sequencing problem based on symmetric homomorphic encryption. The result is of value not only in theory, but also in practice.

## 1. Introduction

Sequencing problem is very common in our daily life, such as ranking according to the scores, queuing by the height. Informally speaking, it is about comparing and sequencing of some numbers. It is easy and convenient to get the result because it cares nothing about privacy in the scenes above. On the contrary, privacy-preserving sequencing problem (PPSP) is always a hard challenge since it requires to conduct secret numbers comparison without knowing the numbers. In this scenario, all participants distrust each other and would not like to leak their own secret information to anyone else. It is an urgent task to be solved for some important applications such as electronic bidding, anonymous voting, and online auction. Naturally, as a powerful tool in solving privacy-preserving cooperative problems, secure multiparty computation (SMC) [1] is the best choice for privacy-preserving sequencing. In fact, the classical Millionaire's problem [1–3] is the earliest example of introducing secure multiparty computation into the sequencing problem. More specifically, the millionaire's problem, with the aim to find out which one of the two Millionaires is richer without revealing their net worth, can

be described as comparing two secret numbers in the perspective of sequencing, that is, the 2-party case of PPSP. In this aspect, the case of 2-party sequencing problem has already been resolved along with the advent of the solutions to Millionaire's problem and the presence of other secure two-party computation protocols [4–12]. Due to the limitation of the 2-party case in practice, the general multiparty PPSP becomes the focus in secure multiparty computation recently.

In 1962, Held and Karp [13] put forward a dynamic programming approach to multiparty sequencing problem before the advent of SMC. They concern more about some certain scenarios and aim to design schemes for the special applications such as the traveling-salesman problem. Subsequently, the research on PPSP is rare and mainly about the 2-party case. Currently, Tang et al. [14] have constructed an efficient and secure multiparty computation protocol for PPSP by making use of a secret sharing scheme based on polynomial. It is an important fruit of PPSP since it has indeed realized secure sequencing among distrusted participants. However, the cost is too high in choosing random numbers and transmitting messages. In the case of  $n$  parties with

$t$  adversaries, it needs to choose  $2n \cdot (2t + 1) + n$  polynomials and  $2n \cdot (t+1)$  random numbers. What is more, the transmitted messages are up to  $(2t + 1) \cdot (n - 1) \cdot n + (n - 1) \cdot n + n^2(2t + 1)$  every round.

This paper applies the fast symmetric homomorphic encryption to replace the cumbersome secret sharing based on polynomial. It no longer needs to choose so many polynomials and random numbers. Relevant complexities in computation and communication also have a great improvement. Our result is not only much simpler but also more efficient. In brief, our contributions can be summarized as follows.

- (1) We first introduce symmetric homomorphic encryption to solve the privacy-preserving sequencing problem in secure multiparty computation, which brings less communications and random numbers than the method of secret sharing based on polynomial.
- (2) Our protocol is appropriate for the insecure channel which allows external attackers to eavesdrop and can resist at most  $t < n/2$  adversaries' corruption supposing that any two neighbor parties do not conspire.
- (3) We propose a protocol for the general privacy-preserving sequencing problem, which is suitable for multiple parties to securely determine the order of a given set rather than just two parties such as the simplest sequencing problem-Millionaire's problem, or a special application such as the traveling-salesman problem.

*Organization.* The rest of this paper is organized as follows. In Section 2, we briefly give some related preliminaries. In Section 3, we present the new efficient secure multiparty computation protocol for privacy-preserving sequencing problem over insecure channel. In Section 4, we analyze the proposed protocol in detail including its correctness and privacy. Furthermore, we show the advantages of our protocol in the two aspects of transmitted messages and random numbers. Finally, we summarize our work of this paper in the last section.

## 2. Preliminaries

*2.1. Secure Multiparty Computation.* Secure multiparty computation is dedicated to dealing with the problem of privacy-preserving cooperative computation among distrusted participants. It was first introduced by Yao in 1982 [1] by putting forward the famous Millionaire's problem. Afterwards, SMC has become a research focus in the international cryptographic community, and a mass of research results have been published one after the other [2–12].

Generally speaking, SMC is a method to implement cooperative computation with all participants' private data, ensuring the correctness of the computation as well as not disclosing additional information except the necessary results. Assume that there are  $n$  participants  $P_1, P_2, \dots, P_n$ . Each has a secret, respectively,  $S_1, S_2, \dots, S_n$ . They want to compute the value of a public function  $F(\cdot)$  on  $n$  variables at the point

$(S_1, S_2, \dots, S_n)$ , that is,  $F(S_1, S_2, \dots, S_n)$ . An SMC protocol is dubbed secure if no participant can learn more from the description of the public function and the result of the global calculation than what he can learn from his own information.

*2.2. Homomorphic Encryption.* In this subsection, we introduce a basic tool to design our protocol, the symmetric homomorphic encryption scheme. Allowing for security, the participants usually would not like to directly transmit their original data over insecure channel while interacting with others. They expect that other parties can perform necessary computations on the encrypted version of the data. In this way, they can encrypt their own private information and then transmit it to others without exposing the real data and finally decrypt the information sent back by others to get the target result when completing cooperative computation. To meet this demand, Rivest et al. proposed homomorphic encryption in 1978 [15]. His work sparked the research in this field. A lot of articles have been proposed and widely used in many applications since then. However, the most common homomorphic encryption schemes are mainly asymmetric, for example, ElGamal homomorphic encryption scheme and Paillier' homomorphic encryption scheme.

Although symmetric homomorphic encryption has not been used in PPSP, it is really a promising method for secure multiparty computation while dealing with the problem of privacy-preserving sequencing. The symmetry will bring high efficiency to our solution since symmetric encryption possesses the advantage of being really fast and can be used as often as possible. As illustrated in [16], a block cipher like AES is typically 100 times faster than RSA encryption and 2000 times than RSA decryption, with about 60 MB per second on a modest platform. Stream ciphers are even faster, some of them being able to encrypt/decrypt 100 MB per second or more. Therefore, asymmetric homomorphic encryptions are bound to much slower than the symmetric ones. In this paper, we will employ the superior symmetric homomorphic encryption schemes to construct our protocol.

Generally, an encryption scheme is said to be homomorphic if for any given encryption key  $k$ , the encryption function  $E(\cdot)$  satisfies the following condition:

$$\forall m_1, m_2 \in P, \quad E(m_1 \circ_P m_2) = E(m_1) \circ_C E(m_2), \quad (1)$$

where  $P(C)$  denotes the set of the plaintexts (ciphertexts), and  $\circ_P$  and  $\circ_C$  are the operators in  $P$  and  $C$ .

We say that a scheme is additively homomorphic if we consider addition operators, and it is multiplicatively homomorphic if we consider multiplication operators. Usually, multiplicative homomorphic encryption functions are more efficient than additive homomorphic encryption functions.

Herein, we will use the random symmetric homomorphic encryption function  $E(\cdot)$  in this paper, which satisfies the following property:

$$\forall m_1, m_2 \in Q^+, \quad E(m_1 + m_2) = E(m_1) * E(m_2), \quad (2)$$

where  $E(\cdot)$  is a random function and  $Q$  is the set of rational numbers.

It is easy to deduce that for all  $m \in Q^+$ ,  $r \in Z^+$ ,

$$E(r * m) = E(m)^r. \quad (3)$$

### 2.3. Privacy-Preserving Sequencing Problem

**2.3.1. The Original Problem.** Privacy-preserving sequencing problem is in fact the more universal description of the generalized secret number comparison. To be more specific, there are  $n$  distrusted participants  $P_1, P_2, \dots, P_n$ . Each of them has a private number, respectively,  $S_1, S_2, \dots, S_n$ . The problem is that they hope to rank the  $n$ -array  $(S_1, S_2, \dots, S_n)$  without leaking any information about  $S_1, S_2, \dots, S_n$ . It requires that after executing cooperative computation,  $P_1, P_2, \dots, P_n$  know the size relations of  $S_1, S_2, \dots, S_n$  but no more other information. Formally, we can represent the whole problem as shown in Algorithm 1.

**2.3.2. Equivalent Transformation of the Original Problem.** In this paper, we make use of a useful theorem in the progressing procedure following reference [14] so that we can reduce the initial sequencing problem about the  $n$ -array  $(S_1, S_2, \dots, S_n)$  to the new  $n$ -array  $(S'_1, S'_2, \dots, S'_n)$ , which has the same sequence as  $(S_1, S_2, \dots, S_n)$  and is called as the pseudoarray of  $(S_1, S_2, \dots, S_n)$ . Then  $P_1, P_2, \dots, P_n$  can obtain the sequence of  $S_1, S_2, \dots, S_n$  by directly comparing the pseudoarrays  $(S'_1, S'_2, \dots, S'_n)$  in public. Along with the equivalent transformation of the problem, the aim of secure multiparty computation needs a corresponding change. It no longer has to consider how to deal with the real data  $S_1, S_2, \dots, S_n$  but only needs to securely get the pseudodata  $S'_1, S'_2, \dots, S'_n$ . And then the subsequent work is just a piece of cake.

**Theorem 1.** Arrays  $(S_1, S_2, \dots, S_n)$  and  $(S'_1, S'_2, \dots, S'_n)$  have the same sequence, where  $S'_i = r_1 * S_i + r_2 * S_i^2 + \dots + r_n * S_i^n$ ,  $r_i \geq 0$ ,  $S_i \geq 0$ ,  $i = 1, 2, \dots, n$ .

*Proof.* Given for all  $S'_i, S'_j \in (S'_1, S'_2, \dots, S'_n)$

$$\begin{aligned} S'_i &= r_1 * S_i + r_2 * S_i^2 + \dots + r_n * S_i^n, \\ S'_j &= r_1 * S_j + r_2 * S_j^2 + \dots + r_n * S_j^n. \end{aligned} \quad (4)$$

Then,

$$\begin{aligned} S'_i - S'_j &= (r_1 * S_i + r_2 * S_i^2 + \dots + r_n * S_i^n) \\ &= (r_1 * S_i + r_2 * S_i^2 + \dots + r_n * S_i^n) \\ &\quad - (r_1 * S_j + r_2 * S_j^2 + \dots + r_n * S_j^n) \\ &= r_1 * (S_i - S_j) + r_2 * (S_i^2 - S_j^2) + \dots + r_n * (S_i^n - S_j^n) \\ &= (S_i - S_j) [r_1 + r_2 * (S_i + S_j) \\ &\quad + r_3 * (S_i^2 + S_i * S_j + S_j^2) + \dots \\ &\quad + r_n * (S_i^{n-1} + S_i^{n-2} * S_j + \dots + S_i * S_j^{n-2} + S_j^{n-1})]. \end{aligned} \quad (5)$$

**Input:**  $(S_1, S_2, \dots, S_n)$ ,  $S_i$  is the private number of  $P_i$ ;  
**Output:**  $(l_1, l_2, \dots, l_n)$ ,  $l_i$  is the order of  $S_i$  in the  $n$ -array.

#### ALGORITHM 1

Let

$$\begin{aligned} Q &= r_1 + r_2 * (S_i + S_j) + r_3 * (S_i^2 + S_i * S_j + S_j^2) + \dots \\ &\quad + r_n * (S_i^{n-1} + S_i^{n-2} * S_j + \dots + S_i * S_j^{n-2} + S_j^{n-1}). \end{aligned} \quad (6)$$

Then,  $S'_i - S'_j = (S_i - S_j) * Q$ . As we know that  $r_i \geq 0$ ,  $S_i \geq 0$ ,  $i = 1, 2, \dots, n$ . Therefore,  $Q \geq 0$ . That means, for all  $S'_i, S'_j \in (S'_1, S'_2, \dots, S'_n)$ ,  $S'_i, S'_j$  and  $S_i, S_j$  have the same sequence. Obviously,  $(S_1, S_2, \dots, S_n)$  and  $(S'_1, S'_2, \dots, S'_n)$  have the same sequence.  $\square$

## 3. Proposed Protocol

In this section, we present our protocol. The simplified version of the protocol is briefly illustrated in Figure 1, and the details can be described as follows.

**Initialization.** Assume that there are  $n$  participants  $P_1, P_2, \dots, P_n$ , each  $P_i$  owning a secret number  $S_i$  and a random symmetric homomorphic encryption function  $E_i(\cdot)$ .

**Computation**

- (1)  $P_i$  chooses a random number  $r_i > 0$ ,  $i = 1, 2, \dots, n$ , and computes  $E_i(S_i), E_i(S_i^2), \dots, E_i(S_i^{i-1}), E_i(S_i^{i+1}), \dots, E_i(S_i^n)$  and  $E_i(r_i * S_i^j)$  locally. For  $i = 1, 2, \dots, n$ ,  $j = 1, 2, \dots, n$ ,  $i \neq j$ ,  $P_i$  sends  $E_i(S_i^j)$  to  $P_j$ .
- (2) After receiving  $E_i(S_i^j)$ ,  $P_j$  computes  $E_i(S_i^j)^{r_j}$ ,  $i = 1, 2, \dots, n$ ,  $j = 1, 2, \dots, n$ ,  $i \neq j$ . And then,  $P_j$  transmits  $E_i(S_i^j)^{r_j}$  to  $P_{i+1}$ ,  $i = 1, 2, \dots, n-1$ ,  $j = 1, 2, \dots, n$ ,  $j \neq i+1$ . For  $i = n$ ,  $P_j$  transfers  $E_n(S_n^j)^{r_j}$  to  $P_1$ ,  $j = 2, \dots, n$ .
- (3)  $P_1$  computes  $S''_n = E_n(r_1 * S_n + r_2 * S_n^2 + \dots + r_n * S_n^n)$  and sends  $S''_n$  to  $P_n$ ; For  $i = 1, 2, \dots, n-1$ ,  $P_{i+1}$  computes  $S''_i = E_i(r_1 * S_i + r_2 * S_i^2 + \dots + r_n * S_i^n)$  and sends  $S''_i$  to  $P_i$ .
- (4)  $P_i$  computes  $S'_i = D_i(S''_i)$ ,  $i = 1, 2, \dots, n$  and broadcasts  $S'_i$  to obtain the sequence of the  $n$ -array  $(S_1, S_2, \dots, S_n)$  by comparing the size of the pseudoarray  $(S'_1, S'_2, \dots, S'_n)$ .

## 4. Analysis

In this section, we have an analysis of the proposed protocol in the aspects of security and efficiency. To guarantee that it is a secure multiparty computation protocol, we have to prove that it satisfies correctness and privacy requirements at first.

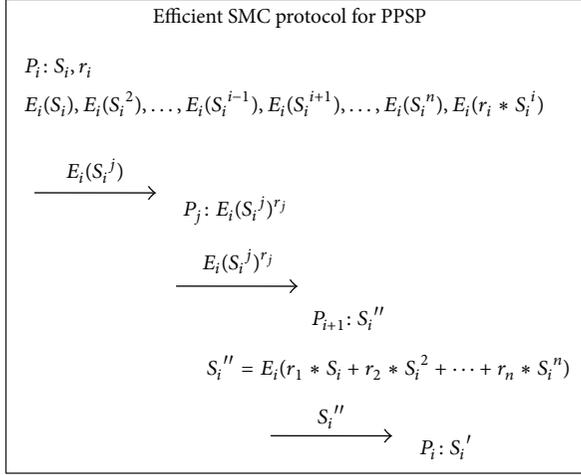


FIGURE 1: Simplified version of the proposed protocol.

**4.1. Correctness.** Assume that the attacker is passive. Then, all participants (including all attackers and honest participants) correctly follow the protocol. Therefore, we only need to examine whether the protocol will give the correct sequence for the array  $(S_1, S_2, \dots, S_n)$ . From the proof of Theorem 1, we know that the array  $(S_1, S_2, \dots, S_n)$  have the same sequence with the pseudoarray  $(S'_1, S'_2, \dots, S'_n)$ . Thus, the proposed protocol can correctly achieve the aim of sequencing the array  $(S_1, S_2, \dots, S_n)$  by comparing the pseudoarrays  $(S'_1, S'_2, \dots, S'_n)$  secretly. Hence, the protocol satisfies correctness.

**4.2. Privacy.** According to the definition of privacy in multi-party computation protocols in [17], the protocol is private if the protocol satisfies the following conditions.

- (a) The information string viewed by each participant  $P_i$  and a random string with the same length have the same probability distribution. That is, the information string and the random string are indistinguishable.
- (b) Arbitrary  $t < n/2$  participants cannot jointly obtain any information about the input of any other participant.

In fact, in the proposed protocol, the viewed information strings of  $P_i$  are  $E_i(S_i), E_i(S_i^2), \dots, E_i(S_i^{i-1}), E_i(S_i^{i+1}), \dots, E_i(S_i^n)$ , and  $E_i(r_i * S_i^i)$  in the first step;  $E_i(r_i * S_i^i), E_j(S_j^i), E_j(S_j^i)^{r_i}, i = 1, 2, \dots, n, j = 1, 2, \dots, n, i \neq j$  in the second step;  $S_{i-1}'', i = 2, \dots, n$ , specially, for  $i = 1$ , the viewed information string in this step is  $S_n''$  for  $P_1$ ; finally, in the last step, the viewed string is  $S_i'$  for  $P_i$ . All the strings are generated by the random symmetric homomorphic encryption function  $E_i(\cdot)$ . Therefore, the strings viewed by  $P_i$  and a random string with the same length have the same probability distribution, and (a) is satisfied.

Moreover, our protocol also satisfies that arbitrary  $t < n/2$  participants cannot jointly obtain any information about the input of any other participant under the assumption that any two neighbor parties never conspire. Since we have  $P_{i+1}$

TABLE 1: Efficiency comparison.

Item	Our protocol	Tang's protocol
Random numbers	$n$	$2n(t+1)$
Transmitted messages	$n(2n-1)$	$(2t+1)(n-1)n + (n-1)n + n^2(2t+1)$

TABLE 2: Efficiency comparison.

Item	Our protocol	Tang' protocol
Random numbers	$n$	$n(n+1)$
Transmitted messages	$n(2n-1)$	$2n^3 - n$

to compute  $S_i''$  for  $P_i$  by collecting  $E_i(S_i^j)^{r_j}, j = 1, \dots, n$ , it is obvious that it is insecure if  $P_i$  and  $P_{i+1}$  collude. It is reasonable to suppose that no neighbors collude because the two adversaries are not exactly adjacent since they cannot control the order of the parties when executing the protocol. In addition, if an adversary wants to get more information from  $S_i'' = E_i(r_1 * S_i + r_2 * S_i^2 + \dots + r_n * S_i^n)$ , he must corrupt with at least  $n-1$  parties since there are  $n-1$  unknown coefficients as well as breaking the encryption scheme  $E(\cdot)$ .

What is more, in our protocol, all information strings are transmitted in the encrypted forms. The private information  $S_i$  and  $r_i$  are secret as long as the encryption function  $E_i(\cdot)$  is robust. In other words, it is secure even over the insecure channel, which is better than the previous protocol based on polynomial for the sequencing problem.

In short, our protocol is correct and private.

**4.3. Efficiency.** Our protocol is efficient as well as secure. It operates better than the previous one because it is independent of the secret sharing scheme based on complex polynomial. We can make a concrete comparison between the proposed protocol and the previous one on the numbers of random numbers and transmitted messages as in Table 1.

From Table 1, we can easily find that in Tang' protocol [14], it needs to choose  $n \cdot (2t+1)$  polynomials for  $f(\cdot)$ ,  $n$  polynomials for  $r(\cdot)$ , and  $n \cdot (2t+1)$  polynomials for  $h(\cdot)$ , that is, totally  $2n \cdot (t+1)$  random numbers as well as  $2n \cdot (2t+1) + n$  polynomials; it also needs to transmit  $f_{ik}(x_j)$  from  $P_i$  to  $P_j, i, j = 1, 2, \dots, n, k = 1, 2, \dots, 2t+1, r_i(x_j), h_{ik}(x_j)$  from  $P_i$  to  $P_j, i = 1, 2, \dots, 2t+1, j, k = 1, 2, \dots, n$ ; thus,  $(2t+1) \cdot (n-1) \cdot n + (n-1) \cdot n + n^2(2t+1)$  messages are needed to be transmitted totally.

In our protocol, it only needs to choose  $n$  random numbers in the whole procedure. And the messages that need to be transmitted are, respectively,  $E_i(S_i^j), i = 1, 2, \dots, n, j = 1, 2, \dots, n, i \neq j, E_i(S_i^j)^{r_j}, i = 1, 2, \dots, n-1, j = 1, 2, \dots, n, j \neq i+1$ , and  $E_n(S_n^j)^{r_j}, j = 2, \dots, n$ , and  $S_i'', i = 1, 2, \dots, n$ , totally  $n \cdot (2n-1)$  messages. It is much simpler and more appropriate for the clients who expect easier products in practice.

If there are  $t = (n-1)/2$  adversaries (the upper bound of the adversaries in Tang' protocol [14]), the advantages of our protocol are more obvious as shown in Table 2.

## 5. Conclusion

It is always a difficult problem in the cryptographic field to construct a secure multiparty computation protocol for the privacy-preserving sequencing problem. In the present study, we have successfully designed an efficient secure multiparty computation protocol for sequencing problem over insecure channel based on symmetric homomorphic encryption, which is of great importance to the theory on this topic and of significant value in practice for its high efficiency.

## Acknowledgments

This work is supported by NSFC (Grant nos. 61170270, 61100203, 60903152, 61003286, and 61121061) and the Fundamental Research Funds for the Central Universities (Grant nos. BUPT2011YB01, BUPT2011RC0505, 2011PTB-00-29, 2011RCZJ15, and 2012RC0612).

## References

- [1] A. C. Yao, "Protocols for secure computations," in *Proceedings of the 23rd Annual IEEE Symposium on Foundations of Computer Science*, pp. 160–164, Chicago, Ill, USA, 1982.
- [2] Y. Lindell and B. Pinkas, "A proof of security of Yao's protocol for two-party computation," *Journal of Cryptology*, vol. 22, no. 2, pp. 161–188, 2009.
- [3] O. S. Goldreich, S. Micali, and A. Wigderson, "How to play any mental game," in *Proceedings of the 19th Annual ACM Symposium on Theory of Computing (STOC '87)*, pp. 218–229, ACM, New York, NY, USA, 1987.
- [4] R. Fagin, M. Naor, and P. Winkler, "Comparing information without leaking it," *Communications of the ACM*, vol. 39, no. 5, pp. 77–85, 1996.
- [5] B. Schoenmakers and P. Tuyls, "Practical two-party computation based on the conditional gate," in *Advances in Cryptology: ASIACRYPT 2004, 10th International Conference on the Theory and Application of Cryptology and Information Security*, pp. 119–136, Springer, Jeju Island, Korea, 2004.
- [6] I. F. Blake and V. Kolesnikov, "Strong conditional oblivious transfer and computing on intervals," in *Advances in Cryptology: ASIACRYPT 2004, 10th International Conference on the Theory and Application of Cryptology and Information Security*, pp. 515–529, Springer, Jeju Island, Korea, 2004.
- [7] Y. L. Luo, *Some key issues in secure multi-party computation and their applied research [Ph.D. dissertation]*, University of Science and Technology of China, 2005 (Chinese).
- [8] M. Fischlin, "A cost-effective pay-per-multiplication comparison method for millionaires," in *Topics in Cryptology: CT-RSA 2001, The Cryptographers' Track at RSA Conference 2001*, pp. 457–471, Springer, San Francisco, Calif, USA, 2001.
- [9] C. Cachin, "Efficient private bidding and auctions with an oblivious third party," in *Proceedings of the 1999 6th ACM Conference on Computer and Communications Security*, pp. 120–127, ACM, New York, NY, USA, November 1999.
- [10] J. Qin, Z.-F. Zhang, D.-G. Feng, and B. Li, "Protocol of comparing information without leaking," *Journal of Software*, vol. 15, no. 3, pp. 421–427, 2004.
- [11] H. Y. Lin and W. G. Tzeng, "An efficient solution to the millionaires problem based on homomorphic encryption," ASIACRYPT 2005, <http://eprint.iacr.org/2005/043>.
- [12] I. Ioannidis and A. Grama, "An efficient protocol for Yao's Millionaires' problem," in *Proceedings of the 36th Hawaii International Conference on System Sciences*, Maui, Hawaii, USA, 2003, Track 7.
- [13] M. Held and R. M. Karp, "A dynamic programming approach to sequencing problems," *Journal of the Society for Industrial and Applied Mathematics*, vol. 10, pp. 196–210, 1962.
- [14] C. Tang, G. Shi, and Z. Yao, "Secure multi-party computation protocol for sequencing problem," *Science China. Information Sciences*, vol. 54, no. 8, pp. 1654–1662, 2011.
- [15] R. L. Rivest, L. Adleman, and M. L. Dertouzos, "On data banks and privacy homomorphisms," in *Foundations of Secure Computation*, pp. 169–179, Academic Press, 1978.
- [16] C. Fontaine and F. Galand, "A survey of homomorphic encryption for nonspecialists," *Eurasip Journal on Information Security*, vol. 2007, Article ID 13801, 2007.
- [17] M. Ben-Or, S. Goldwasser, and A. Wigderson, "Completeness theorems for non-cryptographic fault-tolerant distributed computation," in *Proceedings of the 20th Annual ACM symposium on Theory of Computing (STOC '88)*, pp. 1–11, 1988.

## Research Article

# Study on Semi-Parametric Statistical Model of Safety Monitoring of Cracks in Concrete Dams

Chongshi Gu,<sup>1,2</sup> Dong Qin,<sup>1,2</sup> Zhanchao Li,<sup>1,3</sup> and Xueqin Zheng<sup>1,2</sup>

<sup>1</sup> State Key Laboratory of Hydrology-Water Resources and Hydraulic Engineering, Hohai University, Jiangsu Nanjing 210098, China

<sup>2</sup> College of Water Conservancy and Hydropower Engineering, Hohai University, Jiangsu Nanjing 210098, China

<sup>3</sup> College of Water Resources and Architectural Engineering, Northwest A&F University, Shanxi Yangling 712100, China

Correspondence should be addressed to Dong Qin; qddamsafe@163.com

Received 17 April 2013; Revised 25 June 2013; Accepted 9 July 2013

Academic Editor: Vishal Bhatnagar

Copyright © 2013 Chongshi Gu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Cracks are one of the hidden dangers in concrete dams. The study on safety monitoring models of concrete dam cracks has always been difficult. Using the parametric statistical model of safety monitoring of cracks in concrete dams, with the help of the semi-parametric statistical theory, and considering the abnormal behaviors of these cracks, the semi-parametric statistical model of safety monitoring of concrete dam cracks is established to overcome the limitation of the parametric model in expressing the objective model. Previous projects show that the semi-parametric statistical model has a stronger fitting effect and has a better explanation for cracks in concrete dams than the parametric statistical model. However, when used for forecast, the forecast capability of the semi-parametric statistical model is equivalent to that of the parametric statistical model. The modeling of the semi-parametric statistical model is simple, has a reasonable principle, and has a strong practicality, with a good application prospect in the actual project.

## 1. Introduction

In an investigative report on the state of dams, the International Committee on Large Dams points out that the majority of concrete dams in various countries have cracks, in which a total of 243 dams have collapsed due to these cracks [1]. Cracks are one of the hidden dangers in concrete dams, which are greatly dangerous to their safety. Serious cracks will deteriorate the strength and stability of a dam's body structure, damage its integrity and antipermeability, accelerate the carbonization and corrosion of concrete, and endanger its safe operation [1]. Around the world, there have been many dams whose operations were affected due to cracks, such as the solid gravity dam in Dworshak in America [2]. It has a height of 219 m, and during its construction, there were no serious structural cracks, but after operating for several years, cracks were found on nine of the dam's sections, in which the crack on section 35 was the most serious, with a depth of 50 m, a width of 2.5 mm, and a water percolating capacity of 483 L/s. In addition, the Zillergrund arch dam in Austria, the Zeuzier arch dam in Switzerland, the Revelstoke solid gravity dam in Canada, the Sayano-Shushenskaya gravity arch dam

in Russia, the Koyna gravity dam in India, the Pacoima arc dam in America, and the Sefid Rud buttress dam in Iran were also seriously affected by cracks.

Up to now, the key technologies of concrete dams are still very difficult subjects to research on, with aspects such as evolution law, development trends, and abnormal instability to consider. Crack aperture is an important parameter for determining a crack's development, evolution, and how it would deform a dam's structure. It is significant to establish an appropriate mathematical model in order to accurately determine the aperture of cracks and the changes on their pattern, so as to judge whether these cracks have rendered the dam unstable and to monitor the changes in the concrete dam's structure to prevent accidents from occurring. To this end, Wu and Gu [3] proposed the parametric statistical model of safety monitoring of cracks in concrete dams, with hydraulic pressure, temperature, and timeliness as its main factors, and this method has been widely applied in several projects. However, in affecting a crack's aperture, hydraulic pressure and temperature sometimes are not mutually independent variables. There exists a nonlinear correlation between them, and it is difficult for the parametric statistical model to

separate various components, and the relevant components of such nonlinearity are partially transferred to the residual error. It is thought in this statistical model that flexible crack deformations caused by hydraulic pressure and temperature loads are recoverable, while the crack deformations caused by time are unrecoverable. However, due to hydraulic pressure and temperature, the stress field at the tip of a crack has a singularity, making further irreversible damages. This deformation is not part of the hydraulic pressure and temperature components of this model, but it is part of the residual error component [4]. By further studying the abnormal behaviors of cracks in concrete dams [5–7], it is found that, when such abnormal behavior is observed, mutation occurs to the crack's aperture sequence. In this model, hydraulic pressure, temperature, and timeliness components are the continuous functions of corresponding factors. The combination of hydraulic pressure, temperature, timeliness, and crack aperture components generate a residual error.

Based on the problems stated previously, many scholars have modified and improved the parametric statistical model of safety monitoring of cracks in concrete dams. Reference [8] regarded a concrete dam as an uncertain system and adopted the approximate reasoning model and information distribution method to establish the uncertain model of concrete dam crack aperture. Reference [2] introduced two factors, namely, crack depth and crack initial aperture, to improve the parametric statistical model. Reference [9] studied the influence of crack length on the crack aperture and established the crack abnormality monitoring model in combination with the crack safety monitoring theory and its abnormality diagnosis method. Reference [10] studied the general principle of the statistical and chaotic hybrid forecast model and proposed that such model be used for measuring concrete dam crack aperture. Reference [11] applied the RS rough set theory to simplify the property and sample set of crack monitoring information and the BP neural network for model training to simplify the sample set and proposed a crack aperture monitoring model based on such RS-BP coupling. Reference [12] established a hybrid model of safety monitoring of concrete dam aperture by analyzing the relation between land change and crack aperture. These studies show that the existing models are still based on the parametric statistical model these models do not solve the relevant and nonlinear problems that are posed by the such statistical model, and they are also unable to measure the abnormality of a crack's behavior.

As a result, considering the problems in the parametric statistical model and by introducing the nonparametric component of concrete dam cracks to express the influence in crack openings, this paper establishes the semiparametric statistical model of safety monitoring of cracks in concrete dams to overcome the limitations of the parametric model.

## 2. Construction of Semiparametric Statistical Model of Safety Monitoring of Cracks in Concrete Dams

In the semiparametric model, the function model is divided into three parts, namely, a finite-dimensional parameter

component (trend component), a nonparametric component (abnormal component) belonging to infinite-dimensional function space, and a random error component. The abnormal component represents the unknown component (such as a system error or model deviation) of a functional relationship, and the trend component describes the known ingredients of function in the observed data. For the semiparametric model, Green and Silverman [13] proposed smoothing spline estimation methods, Eubank et al. [14] studied the trigonometric series estimation method, Ivanov and Leonenko [15] studied large sample properties of errors' semiparametric models under long-range dependence, Shi and Chai [16] proposed the basic theory and estimation methods of the semiparametric regression model with the smoothing least squares local polynomial, Zen [17] studied statistical diagnostics with the semiparametric generalized linear model, Hu [18] studied the strong consistency of wavelet estimators in martingale difference using the semiparametric regression model, and Ruppert et al. [19] systematically studied the semiparametric theory and methods. For the theories and methods in kernel estimation, S. Wang and C. Y. Wang [20] studied the semiparametric model kernel estimation when the covariate is lost or when measurement errors occur, Hong [21] studied the data-driven kernel density estimation in the semiparametric regression model, Robinson and Henry [22] studied high-level nuclear semiparametric estimation under long-range dependence, Manzan and Zerom [23] studied kernel density estimation with partial linear additive model, Hjort and Walker [24] studied the kernel density estimation in optimal window width, Haggmann and Scaillet [25] studied bias correction of asymmetric kernel density estimation, and Song [26] studied the moderate bias of deconvolution kernel density estimation when the measurement errors are smooth and are stationarily distributed. Many studies show that, while the semiparametric model is used in solving the complex relationships between estimated parameters and observations, the relative parametric or nonparametric model has obvious advantages which has been widely used in industry, agriculture, economics, medicine, finance, and other fields.

The application of the semiparametric regression model in the field of hydraulic structures safety monitoring is a new research tool and method. Currently, there are only a few published studies on this model. Xu et al. [27] took the abstracted principal components as semiparametric regression parameter components, took the remaining ingredients and model error as unknown nonparametric components using principal component analysis to compensate for the principal component regression, and established the dam deformation monitoring mixed regression model in an attempt to apply the semiparametric regression model into the field of hydraulic structures safety monitoring. This paper combines the research results of existing concrete dam safety monitoring models, focuses on fractured abnormality properties, creates the semiparametric monitoring model, and tests the model's validity.

*2.1. Semiparametric Regression Statistical Model.* In concrete dam cracks, the influence factors are relatively complex. There have been more analyses and studies on the generation

and development mechanism of concrete dam cracks, and researchers have posted the following findings [28]: (1) it is thought that the existence of a crack during the construction period is an important reason for the expansion of cracks at a later stage; (2) it is thought that the unfavorable load combination during the operation of the dam is an important factor for the generation and development of cracks; and (3) it is thought that the cracks are influenced by concrete rheology, crack side plastic aperture, and seepage effects, which are important factors for the generation and expansion of cracks. Therefore, the factors influencing the crack's aperture can boil down to the external load (like hydraulic pressure and temperature) over time. Meanwhile, when the material near the top end of the crack reaches the maximum allowable strain, the crack will be expanded. The modification in the plastic zone is considered, and the formula for the crack aperture is described below [29]:

$$K_0 = \frac{4\sigma}{E} \sqrt{(a + r_p^*)^2 - x^2}, \quad (1)$$

where  $\sigma$  is the strain near the top of the crack,  $E$  is the elasticity modulus,  $a$  is half of the crack's length,  $r_p^*$  is the radius of the plastic zone, and  $x$  is the distance to the top of the crack.

The previous formula states that a concrete dam's aperture is related to  $\sigma$  and is the joint result of external load and crack side plastic deformation. Thus, the statistical model of crack aperture can be expressed as

$$K = K(H) + K(T) + K(\theta) + \varepsilon, \quad (2)$$

where  $K$  is the crack aperture,  $K(H)$  is the component of crack aperture caused by hydraulic pressure factor,  $K(T)$  is the component of crack aperture caused by temperature factor,  $K(\theta)$  is the component caused by timeliness factor, and  $\varepsilon$  is the random error term. The principle for the selection of each component in the formula is as follows.

In the regression model, if the factors influencing the explained variable  $Y$  are divided into two parts, that is,  $X_1, \dots, X_p$  and  $t_1, \dots, t_q$ , according to experience and historical data, it can be thought that  $X_1, \dots, X_p$  are the main parts, and  $Y$  is linear with  $X_1, \dots, X_p$ . While  $t_1, \dots, t_q$  is a certain disturbance factor, the relation between it and  $Y$  is completely unknown, but as its influence is systematical, it is not suitable to include it in the accidental error term. At this moment, if the nonparametric regression model is used for treatment, too much information will be lost. If the linear model is used for treatment, the terms representing error will surely contain systematical components and will no longer conform to the random characteristics of accidental error, thus will make the fitting effect very poor. In this paper, the semiparametric regression model of concrete dam crack openings' monitoring sequence is built.

The general form of the semiparametric model is

$$L = f(X) + g(Z) + \varepsilon, \quad (3)$$

where  $L$  is the response variable,  $X$  and  $Z$  are the relevant concomitant variables,  $\varepsilon$  is the error term,  $f(\cdot)$  is the clear

relation between the observation value and some parameters, which is the parametric component of the model, and  $g(\cdot)$  is the part without a clear function relation between the observation value and the parameter, which is called the non-parametric component of the model.

The semiparametric model is a widely applied statistical model which is deeply focused on the semiparametric regression model. It includes both the parametric regression and the nonparametric regression models. In the semiparametric model, the linear semiparametric model is a kind of a time sequence model that has been used frequently over the recent years. It does not only include a parametric part, but also a nonparametric part. The linear part controls the overall trend of the explained variable, which is applicable for the extension forecast. Meanwhile, the nonparametric part makes a partial adjustment to the explained variable, which makes the model fit better to the sample observation model. Thus, the linear semiparametric model can be expressed as

$$Y = XB + g(t) + \varepsilon, \quad (4)$$

where  $B = (B_1, \dots, B_p)^T$ ,  $p \geq 1$  is the unknown parameter,  $Y = (Y_1, \dots, Y_n)^T$ ,  $X = (X_1, \dots, X_n)^T$ ,  $g = (g(t_1), \dots, g(t_n))$ , and  $g(t_i)$  are the unknown functions defined between  $[0, 1]$ ,  $\varepsilon = (\varepsilon_1, \dots, \varepsilon_n)$  is the random error vector of independent identical distribution, and  $E(\varepsilon_i) = 0$ ,  $E(\varepsilon_i^2) < \infty$ ,  $t_i \in [0, 1]$ ,  $0 \leq t_1 < t_2 < \dots < t_n \leq 1$ ,  $t_i$ , and  $\varepsilon_i$  are the mutually independent parameters.

The estimation problem of the semiparametric model is a point estimation problem of Euclidean space with infinite-dimension nuisance shape parameter, and the estimation methods that are more focused on are the least-square method kernel estimation, spline estimation, partitional polynomial estimation, and trigonometric series estimation [30]. These methods are adopted in this paper to solve the parameter estimation problem of the semiparametric model.

Assuming the probability density kernel function as  $K(\cdot)$  and  $h > 0$  on, for the selected point  $t_k$ , the kernel function  $W_i(t_k)$  is computed as

$$W_i(t_k) = \frac{K((t_k - t_i)/h)}{\sum_{j=1}^n K((t_k - t_j)/h)}, \quad i, j, k = 1, \dots, n. \quad (5)$$

Assume that  $B$  is known, the nonparametric kernel estimation of  $g(t_k)$  is made based on  $\{Y_i - X_i B, t_i\}_{i=1}^n$ :

$$\hat{g}(t_k, B) = \sum_{i=1}^n W_i(t_k) (Y_i - X_i B). \quad (6)$$

The residual error of observation is calculated as  $V_k = X_k B + \hat{g}(t_k, B) - Y_k$ , where  $\hat{g}(t_k, B)$  is expressed as  $V = (I - W)(XB - Y)$  in vector form, and then it is obtained from the least-square criterion:

$$(XB - Y)^T (I - W)^T P (I - W) (XB - Y) = \min. \quad (7)$$

The normal equation of the previous formula is

$$X^T (I - W)^T P (I - W) XB = X^T (I - W)^T P (I - W) Y. \quad (8)$$

If  $X$  is of full rank, then  $X^T(I-W)^T P(I-W)X$  is nonsingular. The least-square estimation of  $B$  is

$$\hat{B} = \left( X^T(I-W)^T P(I-W)X \right)^{-1} X^T(I-W)^T P(I-W)Y. \quad (9)$$

Substituting  $\hat{B}$  into the previous formula gets the estimation  $\hat{g}_k(t, \hat{B}_k)$  of  $g(t)$

$$\hat{g}_k(t, \hat{B}_k) = \sum_{i=1}^n W_i(t) (L_i - X_i \hat{B}_k) = W(Y - X \hat{B}_k). \quad (10)$$

As analyzed previously and in consideration of the parametric statistical model, the semiparametric regression model can be expressed as

$$K = K(H) + K(T) + K(\theta) + g(t) + \varepsilon, \quad (11)$$

where  $H$  is the water depth,  $T$  is the temperature measurement value,  $\theta$  is the time-related influence factor,  $\varepsilon$  is the random error term, and  $g(\cdot)$  is the part without clear function relation between the crack aperture and the parameter, which is called the nonparametric part of the model.

For the nonparametric part  $g(\cdot)$ , it is usually considered to be the systematic error or bias of the model in the general analysis. However, in the semiparametric model, it has a special meaning. Recent studies show [5–7] that concrete dam crack extension will occur if there is an unfavorable load, which means that its occurrence will lead to changes in the crack openings. According to the basic principles of semiparametric model, additional crack openings are generated due to the abnormality of the cracks' behavior, which largely constitutes the nonparametric part of this model. The nonparametric part of the semiparametric model reflects the abnormal characteristics of dam cracks to some extent, which provides a new idea in the abnormality diagnosis of these cracks' behavior.

**2.2. The Validation of Semiparametric Statistical Model.** For the regression analysis, if several nonstationary time series in the model are cointegrated, the regression analysis is still effective, and it is unnecessary to worry about problems caused by nonstationarity, such as spurious regression [26]. Take linear regression with variables  $Y_t = \beta_0 + \beta_1 X_t + \varepsilon_t$  for example, due to  $\varepsilon_t = Y_t - \beta_0 - \beta_1 X_t$ , if  $\{\varepsilon_t\}$  is smooth, then  $\{Y_t - \beta_0 - \beta_1 X_t\}$  is also smooth, which means that either  $Y_t$  and  $X_t$  are stable themselves, or  $Y_t$  and  $X_t$  are single integrations with the same order and have a cointegration relationship. Regression analyses of models under these two cases are valid. Therefore, the model is effective as long as the error sequence is stable. As the stationarity of regression residuals sequence  $\{e_t\}$  and error sequence remains consistent, the stability of  $\{e_t\}$  can be tested by  $\{\varepsilon_t\}$ . Testing the cointegration among the time series can also test the smoothness of their linear regression residuals sequence. The idea of spurious regression and unit root in nonstationary time series compel us to examine the stationarity of regression residuals sequence.

In this paper, the unit root test method is used to analyze the stationarity of the semiparametric model residuals

sequence, which is used for testing the validity of the semiparametric model. First, the regression model of the semiparametric model residuals sequence is established. Second, the characteristic equation of the regression model is extracted. Third, the characteristic roots of the characteristic equation are calculated. Finally, the distance between the characteristic roots and the unit circle are analyzed and compared. If all the characteristic roots are outside of the unit circle, the residual sequence is smooth. Otherwise, the residual sequence is unstable [31–33].

For the residuals sequence  $\varepsilon_1, \varepsilon_2, \dots, \varepsilon_N$  of the semiparametric model, the self-covariance function  $\hat{v}_k$  can be expressed as

$$\hat{v}_k = \frac{1}{N} \sum_{i=1}^{N-k} (\varepsilon_i - \mu)(\varepsilon_{i+k} - \mu), \quad k = 0, 1, 2, \dots, p, \quad (12)$$

where,  $\mu = (1/N) \sum_{i=1}^N \varepsilon_i$ .

Among them, the autocovariance function satisfies

$$\begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_p \end{bmatrix} = \begin{bmatrix} v_0 & v_1 & \dots & v_{p-1} \\ v_1 & v_0 & \dots & v_{p-2} \\ \vdots & \vdots & \ddots & \vdots \\ v_{p-1} & v_{p-2} & \dots & v_0 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix}. \quad (13)$$

According to (13), the moment estimate  $\hat{a}_1, \hat{a}_2, \dots, \hat{a}_p$  of autoregressive coefficient can be obtained, and the moment estimate's formula of white noise variance is computed as follows:

$$\begin{bmatrix} \hat{a}_1 \\ \hat{a}_2 \\ \vdots \\ \hat{a}_p \end{bmatrix} = \begin{bmatrix} \hat{v}_0 & \hat{v}_1 & \dots & \hat{v}_{p-1} \\ \hat{v}_1 & \hat{v}_0 & \dots & \hat{v}_{p-2} \\ \vdots & \vdots & \ddots & \vdots \\ \hat{v}_{p-1} & \hat{v}_{p-2} & \dots & \hat{v}_0 \end{bmatrix}^{-1} \begin{bmatrix} \hat{v}_1 \\ \hat{v}_2 \\ \vdots \\ \hat{v}_p \end{bmatrix}, \quad (14)$$

$$\hat{\sigma}^2 = \hat{v}_0 - \sum_{i=1}^p \hat{a}_i \hat{v}_i.$$

In view of the fact that the order autoregressive model is generally unknown, the order estimation is necessary. When the order is assumed as  $k$ , the corresponding variance  $\hat{\sigma}_k^2$  of fitting residuals can be calculated. Moreover, by using  $\hat{\sigma}_k^2$ , the model order can be determined according to the QHIC criterion:

$$\text{QHIC}(k) = \ln \left( \frac{\hat{\sigma}_k^2}{\hat{\sigma}_k^2} \right) + \frac{2k \ln(\ln(n))}{n} \quad k = 0, 1, 2, \dots, p_0, \quad (15)$$

where  $n$  is the residual number of samples and  $p_0$  is the upper limit of the model order's value. When  $\text{QHIC}(k)$  is at minimum, the corresponding  $k$  is called the QHIC order selection.

After the autoregressive model order and the coefficients of the model of crack residuals sequence are determined,

the characteristic polynomial regression of the autoregressive model and its corresponding characteristic equation can both be obtained. If the autoregressive model residuals sequence of cracks is assumed, its corresponding characteristic equation is

$$1 - \sum_{j=1}^p a_j z^j = 0. \quad (16)$$

The roots of the characteristic equation can be calculated according to formula (16). The minimum root of the characteristic equation is calculated as

$$z_{\min} = \left| \min (z_j = \rho_j e^{i\lambda_j}) \right|, \quad (17)$$

where  $z_j = \rho_j e^{i\lambda_j}$  are all the roots of characteristic equation (16), including possible multiple roots, where  $j = 1, 2, \dots, p$  determine the closest distance between the root of the characteristic equation and the unit circle.

The closest distance between the root of the characteristic equation and the unit circle must be determined. If  $z_{\min} > 1$ , the residuals sequence is smooth and the semiparametric model is proved to be effective; then we model and analyze the cracks data sequence using the semiparametric model.

### 3. Project Case Analysis

The crack near the elevation of 105 m at the lower stream of a gravity arch dam is analyzed based on the semiparametric model. This dam is a concrete gravity dam of concentric circle with variable radiuses. The dam's top elevation is 126.3 m, its maximum height is 76.3 m, and it has 28 sections from left to right. The construction of the dam took 12 years, which was executed in three phases. During the casting of concrete in Phase II, the layer was raised quickly and the interval time of casting the layer was short. The shrinkage distortion of concrete in Phase II was strongly constrained by the concrete in Phase I, causing a crack at the top, which extended from dam block number 5 to number 28, with a length of over 300 m. Upon detection, the crack was up to 5 m deep. Due to this, a cross-dam reinforcing grouting treatment was conducted from dam block number 14 to number 20 in 1973. In 1987, the crack was re-treated with epoxy grouting. The 1680 crack mouth opening displacement data monitored by the crack meter embedded in dam block number 18 from October 10, 1974 to December 18, 2006 were analyzed. According to the parametric statistical model, the CMOD statistical model form of the crack aperture can be expressed as

$$\text{CMOD} = \sum_{i=0}^{m_1} a_i H^i + \sum_{i=1}^{m_2} b_i T_i + (c_1 \theta + c_2 \ln \theta) + \varepsilon. \quad (18)$$

According to the semiparametric regression model, the CMOD semiparametric regression model of the crack aperture can be expressed as

$$\text{CMOD} = \sum_{i=0}^{m_1} a_i H^i + \sum_{i=1}^{m_2} b_i T_i + (c_1 \theta + c_2 \ln \theta) + g(t) + \varepsilon. \quad (19)$$

The basic idea of the regression model is to use the explaining variable included in the model to explain the change in the dependent variable as extensive as possible. Without a reference standard or a guiding policy, it cannot be determined if the model that is selected for the empirical analysis is proper. Generally, a good model should meet the following standards: time saving, identifiable, goodness-of-fit, and consistency in its theoretical and forecasting abilities [34]. The fitting degree of goodness-of-fit can be measured by the calibrated sample determination coefficient  $R^2$  and  $F$  values, which are defined as  $R^2 = \text{ESS}/\text{TSS}$ ;  $F = [R^2/(k-1)]/[(1-R^2)/(n-k)]$ , where ESS is the regression sum of squares, TSS is the total sum of squares,  $n$  is the sample capacity, and  $k$  is the explaining variable and the intercept. It can be seen from the definition of  $R^2$  that the closer  $R^2$  is to 1, the better the resulting regression fitting will be. It can be understood from the definition of  $F$  that  $F$  and  $R^2$  change in the same direction. In other words, the higher the values of  $F$  and  $R^2$  are, the better the model fitting will be.

By applying the Akaike information standard or AIC, and the Schwarz information standard or SIC, the forecast effect of the regression model can be tested. These two information standards are defined as  $\text{AIC} = \exp(2k/n)(\sum e_i^2)/n$  and  $\text{SIC} = n^{k/n}(\sum e_i^2)/n$ , in which  $n$  is the sample capacity,  $k$  is the number of variables in the model, and  $\sum e_i^2$  is the residual sum of squares of the sample. Based on the definitions of AIC and SIC, the previous factors can be deemed as penalty factors to the degree of freedom. The more explaining variables are in the model, the heavier the penalty will be. Relatively, the SIC standard has a heavier penalty than the AIC standard to the degree of freedom. For AIC and SIC, in terms of forecast, the lower the measuring value is, the better the model's forecast will be.

Efficiency, fairness, and consistency are then tested. Efficiency is the ratio between the forecasted value and actual value, so the more the evaluation ratio is approximate to 1, the higher the efficiency will be. Fairness is weighed with the coefficient of variation, which is the ratio between the standard deviation and mean value. For the evaluation of different samples, the evaluation ratio should be the same as much as possible. Consistency means that the evaluation ratio of different samples should not generate progression or regression and that the evaluation result should be consistent with the actual value [35].

According to the solution method of the parametric statistical model, the measured and fitted CMOD are shown in Figure 1, and the residual CMOD curve is shown in Figure 2. Similarly, for the semiparametric statistical model, the measured and fitted CMOD curves are shown in Figure 3, and the residual CMOD curve is shown in Figure 4. See Figures 5 and 3 for the parametric and nonparametric CMOD curves, and see Table 2 for the values of regression and fitted goodness coefficients.

It can be seen that (1) in the regression fitting effect to the original crack aperture, the fitted goodness parameters  $F$  and  $R$  of the semiparametric regression model are  $1.256E+04$  and  $0.991$ , respectively, while those of the parametric regression model (Table 1) are  $1.448E+03$  and  $0.926$ , respectively. Thus,

TABLE 1: Coefficients of the regression and the fitted goodness (parametric model).

Coefficient	$a_0$	$a_1$	$a_2$	$a_3$	$a_4$	$b_1$	$c_1$	$c_2$	$F$	$R$
Value	0.000E + 00	-7.071E - 01	2.204E - 02	-2.253E - 04	7.585E - 07	-4.092E - 02	9.108E - 04	7.861E - 01	1.448E + 03	0.926

TABLE 2: Coefficients of the regression and the fitted goodness (semiparametric model).

Coefficient	$a_0$	$a_1$	$a_2$	$a_3$	$a_4$	$b_1$	$c_1$	$c_2$	$F$	$R$
Value	0.000E + 00	-6.505E - 01	2.032E - 02	-2.080E - 04	7.001E - 07	-3.781E - 02	8.068E - 04	7.911E - 01	1.256E + 04	0.991

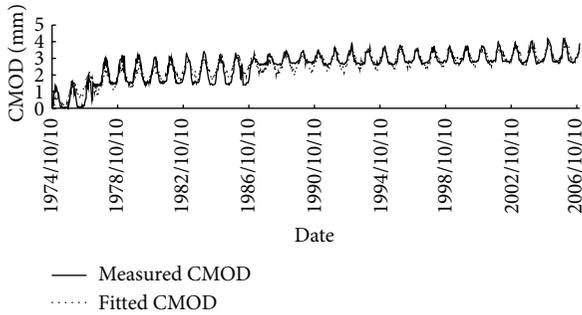


FIGURE 1: Comparison curves of measured and fitted CMOD (parametric model).

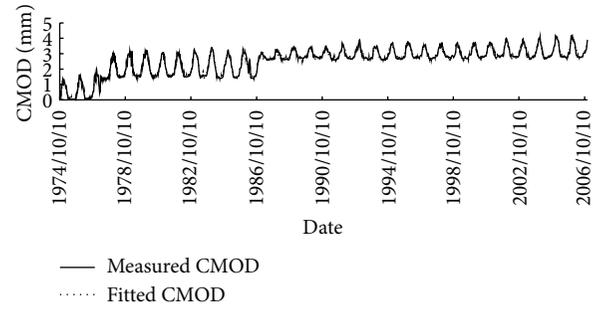


FIGURE 3: Comparison curves of measured and fitted CMOD (semiparametric model).

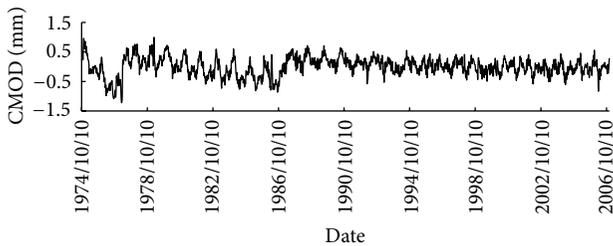


FIGURE 2: Residual CMOD curves (parametric model).

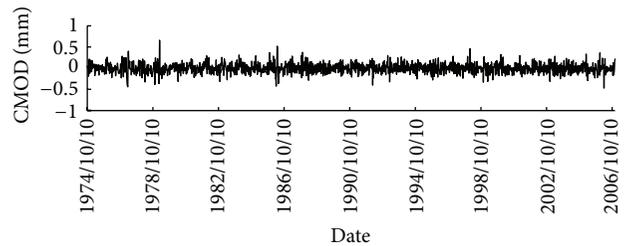


FIGURE 4: Residual CMOD curves (semiparametric model).

the semiparametric model is superior to the parametric regression model; (2) in the characteristics of the parametric sequence of crack aperture, the residual process of the parametric regression model clearly indicates that the residual error does not display a random model, but a certain system model, that is, the error presents systematicness in size and symbol, in Figure 2, it presents a relatively strong periodicity. While the randomness of residual process of the semiparametric regression model is relatively significant, it can be seen that the semiparametric regression model has a stronger explaining ability than the parametric regression model to the original crack aperture.

In order to test the validity of the semiparametric statistical model in the safety monitoring of concrete dam cracks, the residual sequence is segmented and the characteristic roots of each period are calculated. As shown in Figure 7, all minimum eigenvalues of each period are greater than 1.0, indicating that the semiparametric model residuals sequence is stationary; that is, the semiparametric model satisfies the test of validity, the regression analysis on crack sequence can be conducted by using the semiparametric model, and the related forecast monitoring can be carried out.

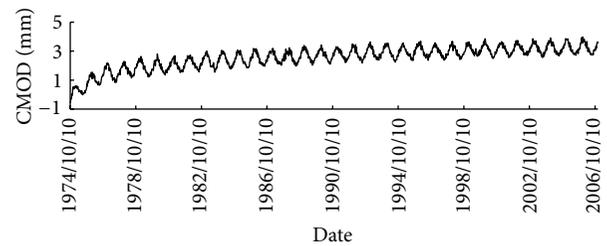


FIGURE 5: Parametric component curves of the CMOD (semiparametric model).

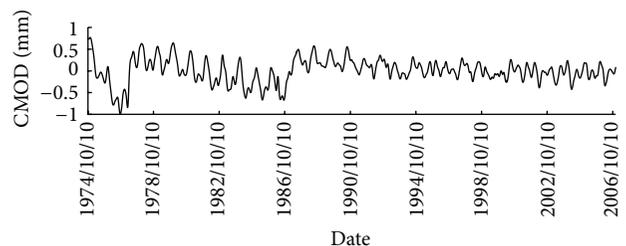


FIGURE 6: Nonparametric component curves of the CMOD (semiparametric model).

TABLE 3: Predicting outcomes of statistical model and Semiparametric regression model.

Date	Measured CMOD (mm)	Predicted CMOD (mm)							
		Semiparametric regression model	Parametric statistical model [28]	uncertainty model [8]	Improved parametric statistical model [2]	Abnormality monitoring model [9]	Hybrid prediction model of statistics and chaos [10]	Monitoring model with RS-BP blending [11]	Hybrid model based on XFEM (extended finite element method) [12]
2005-1-1	4.17	3.92	4.03	4.25	4.06	3.97	3.89	4.10	4.32
2005-6-16	2.82	3.11	3.12	2.63	2.73	2.69	2.97	2.80	2.96
2005-10-20	3.09	3.21	2.97	3.25	3.23	2.86	2.97	3.23	2.79
2006-3-16	3.48	3.36	3.16	3.17	3.38	3.26	3.53	3.60	3.31
2006-6-8	2.79	2.96	2.53	2.56	2.68	2.81	2.91	2.59	2.63
2006-9-1	2.83	2.91	2.68	2.79	2.96	2.79	2.91	2.69	2.81
2006-11-9	3.25	3.35	3.42	3.08	3.37	3.16	3.29	3.17	3.32
2006-12-14	3.90	3.97	4.11	4.12	3.78	3.82	4.03	3.89	4.06
Information standards AIC		0.057	0.052	0.053	0.055	0.056	0.054	0.053	0.058
Information standards SIC		0.070	0.064	0.068	0.069	0.071	0.069	0.067	0.072

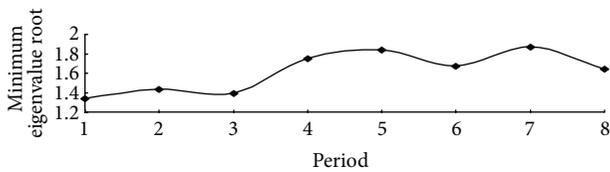


FIGURE 7: The curve of the minimum eigenvalue root in each period.

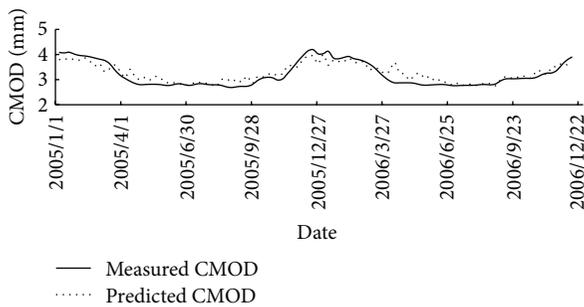


FIGURE 8: The prediction curve of the CMOD (semiparametric model).

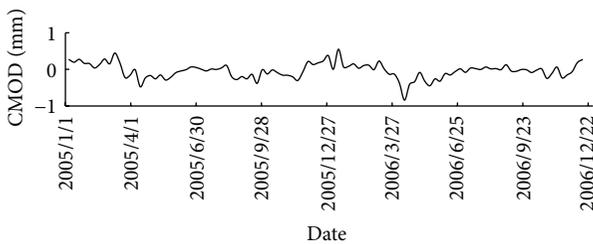


FIGURE 9: Residual curve of the predicted CMOD (semiparametric model).

In order to analyze the forecast effect of the semiparametric statistical model, select crack openings in the time period from 2005-1-1 to 2006-12-22 were used as predictive samples. The measured values are compared and predicted (see Figure 8), and the process curve of prediction error sequence is shown in Figure 9. Meanwhile, the predicted outcomes of the semiparametric model in typical moments are shown in Table 3. We can see from Figures 8 and 9 and Table 3 that AIC and SIC amounted to 0.055 and 0.070, respectively, indicating that the forecast effect of these models is good. In particular, for the parametric regression model, its information standards are, respectively, at 0.052 and 0.064, while the information standards of the semiparametric regression models are, respectively, at 0.057 and 0.070. The semiparametric regression model is used for forecast, but the forecast is only based on the parameters, so the forecast effect of the model is relatively poor. However, the information standards of the two models are approximate. It can be seen that the forecast of the parametric regression model is equivalent to that of the semiparametric statistical model without significant difference.

#### 4. Conclusion

(1) Cracks are one of the hidden dangers in concrete dams, and studying the safety monitoring model of concrete dam cracks has always been a hot topic in such field. The current crack safety monitoring model is the parametric statistical model, which is implemented without fully considering the nonlinear relation between each factor and the irreversible deformation that is caused by hydraulic pressure and temperature loads. The residual error contains plenty of information on both concrete development and evolution.

(2) On the basis of the parametric statistical model and by introducing the nonparametric component of concrete dam

aperture to explain the parts that are imperfectly explained by the parametric model, the semiparametric statistical model of safety monitoring of cracks in concrete dams is established to consider linearity-related factors and to overcome the limitation of the parametric model in expressing the objective model, making this statistical model more appropriate for the objective practice and enabling it to separate system error and accidental error from the error, providing richer calculations for the safety monitoring of cracks in concrete dams.

(3) The semiparametric statistical model is applied in a project, and the result shows that the fitted goodness of the semiparametric statistical model is superior to that of the parametric statistical model. Meanwhile, in the characteristics of the parametric sequence of crack aperture, the residual curve of the parametric statistical model clearly shows that the residual error does not show a random model but a certain system model, while the randomness of the residual curve of the semiparametric statistical model is relatively significant; that is, the semiparametric statistical model has a stronger explaining ability of the original crack aperture than the parametric statistical model. In terms of forecast ability, when the forecast is only based on the parameter of the nonparametric statistical model, its forecast effect is equivalent to that of the parametric statistical model without showing any significant difference.

(4) The semiparametric statistical model is modeled on a simple and reasonable principle and a strong practice, laying a theoretical foundation for the safety monitoring of cracks in concrete dams, with good application prospects in the actual project.

(5) In this paper, studies of the semiparametric statistical model in crack mouth opening displacement have been conducted tentatively, but there are still many issues that need to be studied, such as the following. (1) the parametric section of the semiparametric model is used in predicting crack mouth opening displacement in this paper, and its predictions are the same with the predictions of the parametric statistical model. Therefore, the method of overall prediction using the semiparametric model can become an important research subject in the future; (2) based on this paper's findings, the semiparametric statistical model has a better regression effect in analyzing crack mouth opening displacement. To some extent, the state of abnormality in cracks behavior is reflected in the nonparametric part of the semiparametric model. Thus, the study of the abnormality problem in cracks behavior based on the semiparametric model will also become an important research subject in the future.

## Acknowledgments

This work was supported by Program Sponsored for Scientific Innovation Research of College Graduate in Jiangsu province (Grant no. CXZZ12.0248), the Program for New Century Excellent Talents in University (Grants nos. NCET-11-0628, NCET-10-0359), the Open Foundation of State Key Laboratory of Hydrology-Water Resources (Grant no. 2012490211), the Fundamental Research Funds for the Central Universities (Grant no. 2012BSJJ007), and the Key Project of the National Natural Science Foundation of China (Grant no. 51139001).

## References

- [1] ICOLD, *World Register of Dams*, International Commission on Large Dams, Paris, France, 1988.
- [2] Z. Li, J. Wang, T. Bao, and H. Hou, "Improved safety monitoring model of crack in concrete dams," in *Proceedings of the 12th International Conference on Engineering, Science, Construction, and Operations in Challenging Environments—Earth and Space*, pp. 559–566, ASCE, March 2010.
- [3] Z. Wu and C. Gu, *Safety Monitoring Theories of Hydraulic Structures and Their Application*, Hihger Education Press, Beijing, China, 2003.
- [4] T. Bao, *Chaotic Characteristics of Cracks in Concrete Dam and Their Analysis Theory and Method*, Hohai University, Nanjing, China, 2004.
- [5] X. Li, H. Xu, C. Gu, and Z. Wu, "Detecting abnormality point of dam crack based on phase plane," *Journal of Basic Science and Engineering*, vol. 15, no. 2, pp. 177–182, 2007.
- [6] T. Bao and H. Yu, "Detection of subcritical crack propagation for concrete dams," *Science in China, Series E: Technological Sciences*, vol. 52, no. 12, pp. 3654–3660, 2009.
- [7] C. Gu, Z. Li, and B. Xu, "Abnormality diagnosis of cracks in the concrete dam based on dynamical structure mutation," *Science China Technological Sciences*, vol. 54, no. 7, pp. 1930–1939, 2011.
- [8] F. J. Chen, T. Na, and E. F. Zhao, "Uncertainty analysis model of deformation of cracks in concrete dam," *Journal of Heilongjiang Hydraulic Engineering Collee*, vol. 36, no. 1, pp. 54–60, 2009.
- [9] T. Bao, D. Qin, X. Zhou, and G. Wu, "Abnormality monitoring model of cracks in concrete dams," *Science China Technological Sciences*, vol. 54, no. 7, pp. 1914–1922, 2011.
- [10] T. Bao and Z. Wu, "Hybrid prediction model of concrete crack openings," *Water Resources and Hydropower Engineering*, 2005.
- [11] P. Lei and F. Xiao, "Crack monitoring model based on RS-BP blending," *Journal of Changsha University of Science and Technology (Natural Science Edition)*, vol. 8, no. 2, pp. 45–50, 2011.
- [12] D. J. Zheng, Z. Y. Huo, and B. Li, "Arch-dam crack deformation monitoring hybrid model based on XFEM," *Science China Technological Sciences*, vol. 54, no. 10, pp. 2611–2617, 2011.
- [13] P. J. Green and B. W. Silverman, *Nonparametric Regression and Generalized Linear Models. A Roughness Penalty Approach*, vol. 58 of *Monographs on Statistics and Applied Probability*, Chapman & Hall, London, UK, 1994.
- [14] R. L. Eubank, J. D. Hart, and P. Speckman, "Trigonometric series regression estimators with an application to partially linear models," *Journal of Multivariate Analysis*, vol. 32, no. 1, pp. 70–83, 1990.
- [15] A. V. Ivanov and N. N. Leonenko, "Semiparametric analysis of long-range dependence in nonlinear regression," *Journal of Statistical Planning and Inference*, pp. 1733–1753, 2008.
- [16] Y. C. Shi and G. X. Chai, "Local polynomial smoothing of semiparametric regression model," *Journal of Tongji University*, vol. 28, no. 1, pp. 80–83, 2000.
- [17] L. R. Zen, *Statistical Diagnosis of Semi-Parametric Generalized Linear Models*, East China Normal University, Shanghai, China, 2004.
- [18] H. L. Hu, "Strong consistency of wavelet estimation in semi-parametric regression model," *Chinese Version of Mathematics*, vol. 49, no. 6, pp. 1418–1424, 2006.
- [19] D. Ruppert, M. P. Wand, and R. J. Carroll, *Semiparametric Regression*, vol. 12 of *Cambridge Series in Statistical and Probabilistic Mathematics*, Cambridge University Press, Cambridge, UK, 2003.

- [20] S. Wang and C. Y. Wang, "A note on kernel assisted estimators in missing covariate regression," *Statistics & Probability Letters*, vol. 55, no. 4, pp. 439–449, 2001.
- [21] S.-Y. Hong, "Normal approximation rate and bias reduction for data-driven kernel smoothing estimator in a semiparametric regression model," *Journal of Multivariate Analysis*, vol. 80, no. 1, pp. 1–20, 2002.
- [22] P. M. Robinson and M. Henry, "Higher-order kernel semiparametric  $M$ -estimation of long memory," *Journal of Econometrics*, vol. 114, no. 1, pp. 1–27, 2003.
- [23] S. Manzan and D. Zerom, "Kernel estimation of a partially linear additive model," *Statistics & Probability Letters*, vol. 72, no. 4, pp. 313–322, 2005.
- [24] N. L. Hjort and S. G. Walker, "A note on kernel density estimators with optimal bandwidths," *Statistics and Probability Letters*, vol. 54, no. 2, pp. 153–159, 2001.
- [25] M. Hagmann and O. Scaillet, "Local multiplicative bias correction for asymmetric kernel density estimators," *Journal of Econometrics*, vol. 141, no. 1, pp. 213–249, 2007.
- [26] W. Song, "Moderate deviations for deconvolution kernel density estimators with ordinary smooth measurement errors," *Statistics & Probability Letters*, vol. 80, no. 3–4, pp. 169–176, 2010.
- [27] C. Xu, D.-J. Yue, Y.-F. Dong, and C.-F. Deng, "Regression model for dam deformation based on principal component and semi-parametric analysis," *Yantu Lixue/Rock and Soil Mechanics*, vol. 32, no. 12, pp. 3738–3742, 2011.
- [28] C. S. Gu and Z. R. Wu, *Safety Monitoring Theories and Methods of Dam and Dam Foundation and Their Application*, Hohai University Press, Nanjing, China, 2006.
- [29] A. A. Wells, "Applications of fracture mechanics at and beyond general yielding," *British Welding Journal*, vol. 10, pp. 563–570, 1963.
- [30] S. L. Zhang, *Study on Theory and Application of Least Square Estimation of Nonlinear Semi-Parametric Model*, Wuhan University, Wuhan, China, 2003.
- [31] F. Q. Li, *Analysis Method of Dam Safety Monitoring Data*, Zhejiang University, Hangzhou, China, 2012.
- [32] J. Qian, F. Q. Li, and Y. Zhang, "A study on dam deformation monitoring index based on wavelet transformation and characteristic polynomial root of an auto-regression model," *China Civil Engineering Journal*, vol. 42, no. 6, pp. 140–144, 2009.
- [33] F. Q. Li and J. Qian, "Application of characteristic polynomial roots of autoregression time-series model in analysis of dam observation data," *Journal of Zhejiang University*, vol. 43, no. 1, pp. 193–196, 2009.
- [34] G. Lati, *Econometrics*, Renmin University Press, Beijing, China, 2000.
- [35] J. Jikun, *Study on Application of Nonparametric Regression in the Hedonic Price Evaluation Method of Commercial Residential Buildings*, Zhejiang University, Hangzhou, China, 2006.

## Research Article

# A Hybrid Bat Algorithm with Path Relinking for Capacitated Vehicle Routing Problem

Yongquan Zhou,<sup>1,2</sup> Jian Xie,<sup>1</sup> and Hongqing Zheng<sup>1</sup>

<sup>1</sup> College of Information Science and Engineering, Guangxi University for Nationalities, Nanning, Guangxi 530006, China

<sup>2</sup> Guangxi Key Laboratory of Hybrid Computation and IC Design Analysis, Guangxi University for Nationalities, Nanning, Guangxi 530006, China

Correspondence should be addressed to Yongquan Zhou; [yongquanzhou@126.com](mailto:yongquanzhou@126.com)

Received 8 May 2013; Accepted 24 July 2013

Academic Editor: Praveen Agarwal

Copyright © 2013 Yongquan Zhou et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The capacitated vehicle routing problem (CVRP) is an NP-hard problem with wide engineering and theoretical background. In this paper, a hybrid bat algorithm with path relinking (HBA-PR) is proposed to solve CVRP. The HBA-PR is constructed based on the framework of continuous bat algorithm; the greedy randomized adaptive search procedure (GRASP) and path relinking are effectively integrated into bat algorithm. Moreover, in order to further improve the performance, the random subsequences and single-point local search are operated with certain loudness (probability). In order to verify the validity of the method in this paper, and its efficiency and with other existing methods, several classical CVRP instances from three classes of CVRP benchmarks are selected to tested. Experimental results and comparisons show that the HBA-PR is effective for CVRP.

## 1. Introduction

The vehicle routing problem (VRP) is a classical combinatorial optimization problem that was proposed in the late 1950s, and it is still a research hotspot in the field of Operations Research. The capacitated vehicle routing problem (CVRP) is introduced by Dantzig and Ramser in 1959 [1], which designs a set of customer demands that have to be served with a fleet of vehicles from a depot or central node, each vehicle has the uniform capacity, and each customer has a certain demand that must be satisfied at minimal cost. These costs usually represent distances, traveling times, number of vehicles employed, or a combination of these factors.

It is known that the CVRP is an NP-hard problem [2]. Various approaches have been presented to solve the CVRP during the last decades, such as linear programming [3], several metaheuristics [4–6], and many hybrid heuristics with variable neighborhood search or constructive heuristic methods [7–10]. The overview of methods presented in [6] shows that at least 29 different methods for the CVRP exist; all achieve more or less comparable performance. Although several methods can produce good solutions, the computational time is long when the scale of instances is large. Meanwhile,

an abundance of methods for the CVRP is a population-based algorithm and the parameter setting of the algorithm, is pretty important; however, the parameter setting of many metaheuristics is not considered in the literature.

Bat algorithm (BA) is a fairly new metaheuristics proposed by Yang [11] in 2010, which inspired by the intelligent echolocation behavior of microbats when they forage. As we know, many new metaheuristics have been widely used and successfully applied to solve the CVRP notwithstanding BA has not yet been applied to solve the CVRP, and BA has been applied to solve other problems with great success. For example, Gandom et al. focus on solving constrained optimization tasks [12]. Yang and Gandom apply bat algorithm to solve many global engineering optimizations [13]. A classification mode is proposed by Mishra et al. using bat algorithm to update the weights of a functional link artificial neural network (FLANN) classifier [14]. Meanwhile, some researchers have improved bat algorithm and applied it to various optimization problems. Xie et al. proposed a DLBA bat algorithm based on differential operator and Lévy flights trajectory to solve function optimization and nonlinear equations [15]. Wang et al. proposed a new bat algorithm with mutation (BAM) to solve the uninhabited

combat air vehicle (UCAV) path planning problem [16]. In this paper, we propose a hybrid bat algorithm (HBA-PR) to solve capacitated vehicle routing problem.

The rest of this paper is organized as follows. The problem description of CVRP and original bat algorithm are described in Section 2. The hybrid bat algorithm (HBA-PR) for CVRP is described detailedly in Section 3. The experimental results of the HBA-PR and comparisons with other previous algorithms are shown in Section 4. In the last section, we conclude this paper and point out some future work in Section 5.

## 2. Problem Descriptions and Bat Algorithm

**2.1. Capacitated Vehicle Routing Problem.** The CVRP is considered to be the classical version of the VRP, which designs a set of customer demands that have to be served with a fleet of vehicles from a depot or central node, each vehicle has the uniform capacity, and each customer has a certain demand. The objective makes the expended cost as minimal as possible. Let  $G = (V, E)$  be a complete graph with a set of vertices  $V = \{0, 1, \dots, k\}$ , where the vertex  $\{0\}$  represents the depot and the remaining ones represent the customers. Each edge  $e_{ij} = \{i, j\} \in E$  has a nonnegative cost  $c_{ij}$ , and each customer  $i \in V' = V \setminus \{0\}$  has a demand  $d_i$ . Let  $S = (1, 2, \dots, m)$  be the set of homogeneous vehicles with capacity  $Q$ . The CVRP consists in constructing a set of up to  $k$  routes in such a way that (1) every route starts and ends at the depot; (2) all demands are accomplished; (3) the vehicle's capacity is not exceeded; (4) a customer is visited by only a single vehicle; (5) the sum of costs is minimized. The mathematical formulas are defined as follows [10]:

$$\begin{aligned} \min \quad & Z = \sum_{i=0}^k \sum_{j=0}^k \sum_{s=0}^m c_{ij} e_{ijs}, \quad (1) \\ \text{s.t.} \quad & \sum_{i=0}^k d_i y_{is} \leq Q, \quad s = 1, 2, \dots, m, \\ & \sum_{s=1}^m y_{is} = 1, \quad i = 1, 2, \dots, k, \\ & \sum_{i=0}^k e_{ijs} = y_{is}, \quad j = 1, 2, \dots, k; \quad s = 1, 2, \dots, m, \\ & \sum_{j=0}^k e_{ijs} = y_{is}, \quad i = 1, 2, \dots, k; \quad s = 1, 2, \dots, m, \end{aligned} \quad (2)$$

where  $s$  denotes the number of vehicles;  $e_{ijs} = 1$  if vehicle  $s$  is from  $i$  to  $j$ , otherwise  $e_{ijs} = 0$ ; the  $y_{is} = 1$  if vehicle  $s$  is loading (active), otherwise  $y_{is} = 0$ .

$$\begin{aligned} \text{R number 1: } & \text{Depot} \rightarrow \textcircled{0} \rightarrow \textcircled{3} \rightarrow \textcircled{5} \rightarrow \textcircled{8} \rightarrow \textcircled{0} \text{ Depot} & \text{R number 2: } & \text{Depot} \rightarrow \textcircled{0} \rightarrow \textcircled{4} \rightarrow \textcircled{9} \rightarrow \textcircled{0} \text{ Depot} & \text{R number 3: } & \text{Depot} \rightarrow \textcircled{0} \rightarrow \textcircled{2} \rightarrow \textcircled{6} \rightarrow \textcircled{1} \rightarrow \textcircled{7} \rightarrow \textcircled{0} \text{ Depot} \end{aligned} \quad (6)$$

the coded individual with integer is  $0 \rightarrow 3 \rightarrow 5 \rightarrow 8 \rightarrow 4 \rightarrow 9 \rightarrow 2 \rightarrow 6 \rightarrow 1 \rightarrow 7 \rightarrow 0$ , and the bat individual

**2.2. Basic Bat Algorithm.** The basic bat algorithm (BA) is a metaheuristic first introduced by Yang in 2010. In simulations of BA, under several under idealized rules, the updated rules of bat's positions  $x_i$  and velocities  $v_i$  in a  $D$ -dimensional search space are defined. The new solutions  $x_i^t$  and velocities  $v_i^t$  at generation  $t$  are given by

$$\begin{aligned} \text{fr}_i &= \text{fr}_{\min} + (\text{fr}_{\max} - \text{fr}_{\min}) \beta, \\ v_i^t &= v_i^{t-1} + (x_i^t - x_*) \text{fr}_i, \\ x_i^t &= x_i^{t-1} + v_i^t, \end{aligned} \quad (3)$$

where  $\beta \in [0, 1]$  is a random vector drawn from a uniform distribution and  $\text{fr}_i$  denotes frequency of each bat. Generally, the frequency  $\text{fr}_i \in [\text{fr}_{\min}, \text{fr}_{\max}]$ . Here  $x_*$  is the current global best location (solution) which is located after comparing all the solutions among all the  $n$  bats.

After the position updating of bat, a random number is generated; if the random number is greater than the pulse emission rate  $r_i$ , a new position will be generated around the current best solutions; it can be represented by

$$x = x_* + \varepsilon \times \text{Ld}_t, \quad (4)$$

where  $\varepsilon \in [-1, 1]$  is a random number, while  $\text{Ld}_t = \langle \text{Ld}_i^t \rangle$  is the average loudness of all the bats at current generation  $t$ .

Furthermore, the loudness  $\text{Ld}_i$  and the pulse emission rate  $r_i$  will be updated, and a solution will be accepted if a random number is less than loudness  $\text{Ld}_i$  and  $f(x_i) < f(x_*)$ .  $\text{Ld}_i$  and  $r_i$  are updated by

$$\text{Ld}_i^{t+1} = \alpha \times \text{Ld}_i^t, \quad r_i^{t+1} = r_i^0 \times [1 - \exp(-\gamma \times t)], \quad (5)$$

where  $\alpha, \gamma$  are constants and  $f(\cdot)$  is fitness function. The algorithm repeat until the termination criterion is reached.

## 3. Hybrid Bat Algorithm with Path Relinking for CVRP

**3.1. Solution Representation in HBA-PR.** Since standard BA is a continuous optimization algorithm, the standard continuous encoding scheme of BA cannot be used to solve CVRP directly. In order to apply BA to solve CVRP, the first step is to devise a suitable representation for the candidate solutions in designing a hybrid bat algorithm for a particular problem. Each individual is a sequence with integer number which is the order of visiting these customers; the 0 represents the depot. For example, if we have the following three routes:

is represented as  $3 \rightarrow 5 \rightarrow 8 \rightarrow 4 \rightarrow 9 \rightarrow 2 \rightarrow 6 \rightarrow 1 \rightarrow 7$ .

**3.2. Hybrid Bat Algorithm.** Aimed at the capacitated vehicle routing problem, based on the idea of bat algorithm, a hybrid bat algorithm is proposed, which integrates greedy randomized adaptive search procedure (GRASP) heuristic and bat algorithm, and the path relinking as an intensification strategy to explore local trajectories connecting elite solutions obtained by the proposed algorithm. The hybrid bat algorithm with path relinking is named as HBA-PR.

GRASP [17, 18] is a heuristic already applied to many optimization problems successfully [19–21]. GRASP consists of a two-phase iterative process: construction phase and local search phase. In the first phase, a greedy randomized solution is built. Due to this solution is not guaranteed to be locally optimal; a local search is performed in the second phase. The final result is simply the best solution found over all iterations.

The construction phase can be described as a process which stepwise adds one element at a time to a partial (incomplete) solution. According to a greedy function, all elements are sorted, and the Restricted Candidate List (RCL) is constructed based on the order, and then from the list randomly select a element. In the second phase, a local search is initialized from these points, this iterative process is repeated until a termination criterion is met, and the best solution found over all iterations is taken as the result.

RCL is created using a parameter  $\alpha$  to restrict the size of this list. Candidate  $e \in C$  is sorted according to their greedy function value  $f(e)$ . In a cardinality-based RCL, the latter is made up by the  $k$  top-ranked elements. In a value-based construction, the RCL consists of the elements in the set  $\{e \in C : f_* \leq f(e) \leq f_* + \alpha \times (f^* - f_*)\}$ , where  $f_* = \min\{f(e) : e \in C\}$ ,  $f^* = \max\{f(e) : e \in C\}$ , and  $\alpha \in [0, 1]$ . Since the best value for  $\alpha$  is often difficult to determine, a random value is often assigned. The values for  $\alpha$  adopted in the constructive heuristics are set using reactive strategies, which usually leads to better performance than using fixed values [22].

In original bat algorithm, the bat individual randomly selects a certain range of frequency, its velocity is updated according to their selected frequency, and at last a new position is generated using its velocity and its own position. The idea is that the position of bat individual is updated by adjusting its frequency of sonic pulse. In this paper, the position updating of bat adopted GRASP to generate a new position, the frequency is used for restricting the size of CRL, frequency equivalent to parameter  $\alpha$  in GRASP, and the frequency is variable value. In basic bat algorithm, the loudness  $L_d$  and the pulse emission rate  $r$  are updated according as the iterations proceed. As the loudness usually decreases, while the rate of pulse emission increases, it indicates that the bats approximate their prey (optimum solution). The pulse emission rate  $r$  is updated by

$$r(t) = \left( 1 + \exp \left( \frac{-5}{t_{\max} \times (t - t_{\max}/2)} \right) \right)^{-1}, \quad (7)$$

where  $t$  denotes the  $t$ th generation and  $t_{\max}$  is the maximal generation. The rate  $r$  is similar to Sigmoid function, and the

frequency  $fr$  is determined according to the pulse emission rate  $r$ , which is represented by

$$fr = \begin{cases} 1 - \max(0.2, \min(0.8, r)), & \text{rand} > r, \\ \max(0.2, \min(0.8, r)), & \text{rand} \leq r, \end{cases} \quad (8)$$

where  $\text{rand}$  is a random number and 0.2 and 0.8 are experience parameters reference [18]. The frequency  $fr$  decreases gradually at firstly and then increases gradually, while the generation  $t$  increases. Figure 1 is the changing curve of rate  $r$ , Figure 2 is an example of frequency  $fr$ , and Algorithm 1 shows the pseudocode of greedy randomized construction with frequency  $fr$ .

The local search phase uses the 2-Opt heuristic for exchanging. Algorithm 2 shows the local search procedure. The input parameter is an initial solution  $S$  obtained by the Greedy\_Randomized\_Construction procedure. The current route need is divided into  $m$  subroute according to the load, set the start, and end of suroute said to 0; for example,  $S = \{1, 2, 3, 4, 5, 6, 7\}$ , subroute is  $0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 0$ , and  $0 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 0$ .

**3.3. Hybrid Bat Algorithm with Path Relinking.** Path relinking was originally proposed by Glover [23]; Laguna and Martí [24] were the first to use path relinking within a GRASP strategy. Path relinking generates new solutions by exploring trajectories connecting an initial solution  $x_s$  to an elite guiding solution  $x_t$ . The path relinking procedure consists in selecting moves that introduce attributes contained in the guiding solution  $x_t$  to the initial solution  $x_s$  until the initial solution is completely transformed in the guiding solution  $x_t$ . Path relinking may also be viewed as a constrained local search strategy applied to the initial solution  $x_s$ . Furthermore, there are several alternatives that have been considered, which involve tradeoffs between computation time and solution quality. These alternatives include *periodical relinking*, *forward relinking*, *backward relinking*, *back and forward relinking*, *mixed relinking*, *randomized relinking*, and *truncated relinking* [18].

One important issue in implementing a path relinking technique is the strategy to construct the elite set (ES). We adopted a fixed size elite set, and a solution  $x$  is inserted into the ES as follows.

A solution  $x$  is always inserted into ES if it is not full. Otherwise, the generated solution  $x$  is inserted in ES only if its cost is better than the worst cost solution found in ES, and the worst cost solution is replaced by the solution  $x$ . Algorithm 3 shows the pseudocode for the algorithm to construct and maintain the elite set ES.

Algorithm 4 shows the pseudocode of path relinking procedure; an instance is  $x_s = \{1, 2, 3, 4, 5, 6, 7\}$ ,  $x_t = \{1, 3, 4, 2, 5, 6, 7\}$ , according to process of Algorithm 4, the first different element is position 2 in  $x_s$ ; after executing replace operation, the  $x_1 = \{1, 3, 4, 2, 5, 6, 7\}$ ; the second different element is position 3 in  $x_s$ ; after executing replace operation, the  $x_2 = \{1, 2, 4, 3, 5, 6, 7\}$ ; the third different element is position 4 in  $x_s$ ; after executing replace operation, the  $x_3 = \{1, 4, 3, 2, 5, 6, 7\}$ ; if the  $x_2$  is better than the  $x_t$ , and the only, then return  $x_2$ .

```

S ← R rand_Chosen_Vertex(v); // R is selected at random a vertex as initial solution
C ← V \ v; // The candidate set C is initialized
While C ≠ ∅ do
    IC ← Evaluate_Incremental_Costs(S); // the incremental costs are evaluated
    icmin ← min ({ic ∈ IC});
    icmax ← max ({ic ∈ IC});
    RCL ← {e ∈ C | ic(e) ≤ icmin + fr × (icmax - icmin)}; // RCL is created
    s ← Select_Element (RCL); // a vertex s is randomly selected from RCL
    S ← Obtain_Min_Solution(S, s); // the partial tour is updated by inserting the
    vertex s
    C ← C \ {s}; // the candidate set C is updated
end
return S

```

ALGORITHM 1: Greedy\_Randomized\_Construction (fr).

```

S' ← S;
{sr} ← Construct_Sub-route(S');
for each sub-route sr do
    sr' ← 2-opt (sr); // carry out the 2-opt operation
end
S' ← Construct_Individual ({sr'});
if f(S') < f(S) then
    S ← S'
end
return S

```

ALGORITHM 2: Local\_Search\_Phase (S).

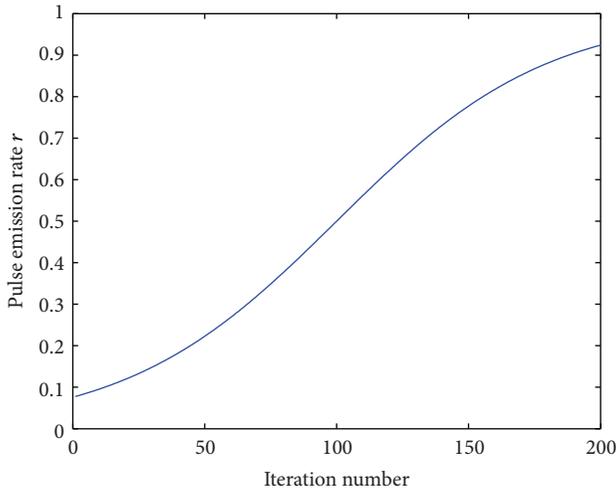


FIGURE 1: Changing curve of rate r.

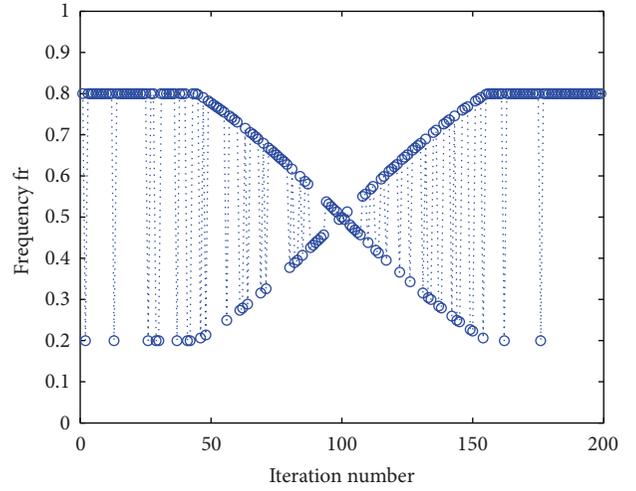


FIGURE 2: Changing curve of frequency fr.

```

if cardinality(ES) ≠ ps then // ps is the population size
    ES ← ES ∪ {x}
else
    ESw ← the worst solution in ES
    if f(x) < f(ESw) then
        ES ← ES \ {ESw}
        ES ← ES ∪ {x}
    end
end
return ES

```

ALGORITHM 3: Elite\_Set (x).

3.4. *Subsequence and Single-Point Local Search.* In this paper, the loudness  $Ld_i$  of bat individual  $i$  is relative to its own fitness  $fit_i$ , the better fitness, and the lower loudness. The loudness can be described by

$$Ld_i = \frac{(fit_i - fit_* + 0.1)}{(fit^* - fit_* + 0.1)}, \quad (9)$$

where constant 0.1 is used for avoiding the denominator is zero, and  $fit_i$  is the fitness of individual  $i$ ,  $fit_*$  and  $fit^*$  are the minimum and maximum fitness in current population, respectively. In HBA-PR, the loudness reflects the quality of individual. In this algorithm, there are two kinds of local search are embedded into HBA-PR to further improve the performance, random subsequence local search that and

```

 $x_s \leftarrow x$ ; // initial solution
 $x_t \leftarrow ES_*$ ; // guiding solution
 $f(x_t) \leftarrow f(ES_*)$ ;
 $\Delta \leftarrow \text{Difference}(x_s, x_t)$ ; // find out the difference position between  $x_s, x_t$ 
for  $i = 1$ :  $\text{cardinality}(\Delta)$  do
   $j \leftarrow \text{Find\_Position}(x_s, x_t, \Delta_i)$ ; // find the position of the element of  $x_t$  in  $\Delta_i$  which in  $x_s$ 
   $x_i \leftarrow \text{Replace}(x_s, \Delta_i, j)$ ; // replace the  $j$  position of  $x_s$  with the element of  $x_s$  in  $\Delta_i$ 
   $x_i \leftarrow \text{Replace}(x_s, x_t, \Delta_i)$ ; //replace the  $\Delta_i$  position of  $x_s$  with the corresponding element of  $x_t$ 
  if  $f(x_i) < f(x_t)$  then
     $x_* \leftarrow x_i$ ;
     $f(x_i) \leftarrow f(x_i)$ ;
  end
end
return  $x_*$ 

```

ALGORITHM 4: Path\_Relinking ( $ES_*$ ,  $x$ ).

single-point local search. The random subsequence local search includes random subsequence inverse and random subsequence insert, and the single-point local search includes single-point insert and single-point swap.

For random subsequence insert, firstly, randomly selected an origin of subsequence, and then randomly selected a length of subsequence which is less than length of individual  $S$ . Secondly, after determining the subsequence  $S1$ , randomly selected an insert point in remainder subsequence  $S2$ ,  $S = S1 \cup S2$ ; the  $S1$  is inserted into  $S2$  location in insert point. An example is shown in Figure 2. For random subsequence inverse, firstly, randomly selected a subsequence with random length, and then the inverse operation is performed. An example is shown in Figure 3.

For single-point swap, choose two different positions from a permutation randomly and swap them. For single-point insert, choose two different positions from a permutation randomly, and the element in first position is insert into the back of second element. Similarly, two instances are shown in Figures 4 and 5.

In local search part, the random subsequence local search is performed before random single-point local search. The random subsequence insert and random subsequence inverse are preformed according to loudness  $Ld$ . In other words, if a random number is greater than the loudness  $Ld_i$ , the random subsequence insert is performed; otherwise, the random subsequence inverse is performed. Similarly, the random single-point insert and random single-point swap are performed with the loudness  $Ld$ . If a random number is greater than the loudness  $Ld_i$ , the insert operation is performed (Figure 6); otherwise, the swap operation is performed. Note that where local search is operated on the current optimal individual  $ES_*$  in elite set  $ES$ , local search is performed for each individual. Algorithms 5 and 6 show the pseudocode of subsequence and single-point local search.

**3.5. HBA-PR Framework for CVRP.** We propose in this work, to incorporate greedy randomized adaptive search procedure, path-relinking strategies, subsequence, and single-point local search to the bat algorithm by defining distinct ways to solve capacitated vehicle routing problem. This iterative process is

repeated until the termination criterion is met; Algorithm 7 shows the pseudocode of HBA-PR for CVRP.

#### 4. Numerical Simulation Results and Comparisons

To test the performance of the proposed HBA-PR which is extensively investigated by a large number of experimental studies computational simulations are carried out with some well-studied problems taken from the web <http://www.branchandcut.org/>, a reference site which contains detailed information regarding a large number of benchmark instances. In this paper, 12 instances from three classes of benchmarks are selected. The first class is Augerat et al. Set A instances, the second class is Augerat et al. Set P instances, and the third class is Augerat et al. Set E instances. So far, these problems have been widely used as benchmarks to certify the performance of algorithms by many researchers.

All computational experiments are conducted with MATLAB 2012a, and in our simulation, numerical experiments are run on a PC with AMD Athlon (tm) II X4 640 Processor 3.0 GHz and 2.0 GB memory. In the experiment, the termination criterion is set as maximum generation  $t_{\max} = 200$ . Each instance independently run 15 times for comparison.

**4.1. Parameter Analysis.** In the subsection, parameters of HBA-PR are determined by experiments, and the impact of each parameter is analyzed. In particular, the HBA-PR has few parameters; we only need to test population size ( $ps$ ) in HBA-PR. A small  $ps$  which may lead to insufficient population information is provided, and the diversity cannot be guaranteed. On the other side, a large one which indicates diversity is sufficient, but the computing time will increase and the precision of optimal solution may have lesser improvement. In order to evaluate the sensitivity of parameters  $ps$ , three benchmarks selected from different benchmark set are chosen to run 10 times. These benchmarks are  $A\_n33\_k5$ ,  $E\_n23\_k3$  and  $P\_n19\_k2$ , and the statistical result and convergence curves are shown in Figures 7, 8, 9, and 10.

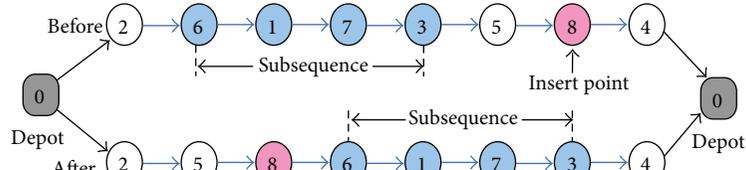


FIGURE 3: Random subsequence insert operation.

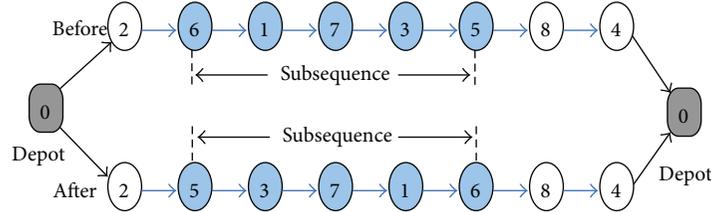


FIGURE 4: Random subsequence inverse operation.

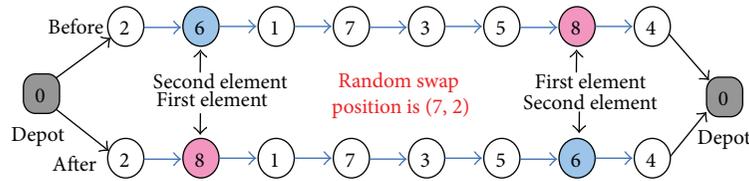


FIGURE 5: Single-point swap operation.

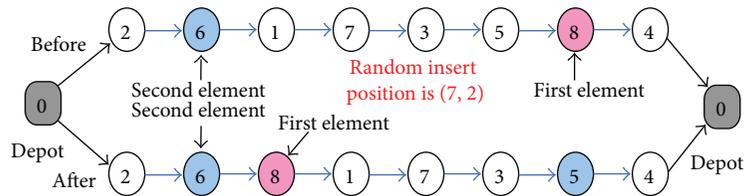


FIGURE 6: Single-point insert operation.

```

for  $i = 1: ps$  do
  if  $\text{rand} > Ld_i$  then
     $x \leftarrow \text{Sub-sequence\_Insert}(ES_*)$ ; // perform random sub-sequence insert operation
  else
     $x \leftarrow \text{Sub-sequence\_Inverse}(ES_*)$ ; // perform random sub-sequence inverse operation
  end
end
return  $x$ 

```

ALGORITHM 5: *Sub-sequence\_Local\_Search* ( $ES_*$ ).

```

for  $i = 1: ps$  do
  if  $\text{rand} > Ld_i$  then
     $x \leftarrow \text{Single-point\_Insert}(ES_*)$ ; // perform random single-point insert operation
  else
     $x \leftarrow \text{Single-point\_Swap}(ES_*)$ ; // perform random single point swap operation
  end
end
return  $x$ 

```

ALGORITHM 6: *Single-point\_Local\_Search* ( $ES_*$ ).

```

(1) Initialize the  $ps$ , bat population and other parameters;
(2) Evaluate fitness for each individual;
(3)  $ES \leftarrow Elite\_Set(x)$ ;
(4) while  $t \leq t_{max}$  do
(5)   Compute pulse emission rate by (7);
(6)   Determine frequency  $fr$  by (8);
(7)   for  $i = 1: ps$  do
(8)      $x \leftarrow Greedy\_Randomized\_Construction(fr)$ ;
(9)      $x \leftarrow Local\_Search\_Phase(x)$ ;
(10)  end
(11)  Evaluate fitness for each individual  $x$ ;
(12)   $ES \leftarrow Elite\_Set(x)$ ;
(13)   $ES_* \leftarrow Select\_Best\_Elite(ES)$ ; // select the best individual in elite set  $ES$ 
(14)  for  $i = 1: ps$  do
(15)     $x \leftarrow Path\_Relinking(ES_*, x)$ ;
(16)  end
(17)  Evaluate fitness for each individual  $x$ ;
(18)   $ES \leftarrow Elite\_Set(x)$ ;
(19)   $Ld_i \leftarrow Compute\_Loudness(f(x))$ ; //Compute loudness of each individual by (9);
(20)   $ES_* \leftarrow Select\_Best\_Elite(ES)$ ;
(21)   $x \leftarrow Sub\_sequence\_Local\_Search(ES_*)$  //carry out random sub-sequence local search
(22)  Evaluate fitness for each individual  $x$ ;
(23)   $ES \leftarrow Elite\_Set(x)$ ;
(24)   $ES_* \leftarrow Select\_Best\_Elite(ES)$ ;
(25)   $x \leftarrow Single\_point\_Local\_Search(ES_*)$  //carry out random single-point local search
(26)  Evaluate fitness for each individual  $x$ ;
(27)   $ES \leftarrow Elite\_Set(x)$ ;
(28)   $t = t + 1$ 
(29) end
(30) Output result and plot

```

ALGORITHM 7: HBA-PR.

TABLE 1: Comparison of results for Augerat et al. Set A, E, and P instances.

Instance	Capacity	Tightness	L.BKS	R.BSK	CWS	SR-GCWS	CS-GRASP	HBA-PR
A-n33-k5	100	0.89	661	662.76	712.05	662.11	662.1452	662.1101
A-n33-k6	100	0.9	742	742.83	776.26	742.69	743.5785	742.6933
A-n37-k5	100	0.81	669	672.59	707.26	672.47	672.4652	672.4652
A-n39-k6	100	0.88	831	833.20	863.08	833.20	—	835.2518
E-n23-k3	4500	0.75	569	—	—	—	569.7461	568.5625
E-n22-k4	6000	0.94	375	375.28	388.77	375.28	—	375.2798
E-n33-k4	8000	0.92	839	838.72	843.1	837.67	—	837.9253
E-n51-k5	160	0.97	521	524.94	584.64	524.61	—	524.6111
P-n19-k2	160	0.97	212	212.66	237.90	212.66	212.6569	212.6569
P-n20-k2	160	0.97	216	217.42	234.00	217.42	217.4156	217.4156
P-n22-k2	160	0.96	216	217.85	239.50	217.85	217.8522	217.8522
P-n51-k10	80	0.97	741	742.48	790.97	741.50	—	743.2648

The ordinate normalized fitness (log) in Figures 8–10 is logarithm of normalized fitness; the aim is to show the convergence curves clearly. The normalization formula of fitness is  $(fit - fit_*) / (fit^* - fit_*)$ , where  $fit_*$  is the best-known solution, and  $fit^*$  is the initial fitness.

Figure 7 represents the relative error of test case  $A\_n33\_k5$ ,  $E\_n23\_k3$ , and  $P\_n19\_k2$  after 10 times independent running, which shows the sensitivity of parameter  $ps$ . From the three

test cases, the parameter  $ps$  can be determined. For  $A\_n33\_k5$  and  $P\_n19\_k2$ , the performance of HBA-PR is better when  $ps = 30$ . For  $E\_n23\_k3$ , that  $ps$  is equal to 50 is best, but the performance is good as well, while  $ps = 30$ . From Figures 8–10, the information provided by population is sufficient when  $ps$  is greater than 20; however, the convergence rate is better when  $ps = 30$  among the three instances. Considering the tradeoff between the stability of algorithm and the rate of

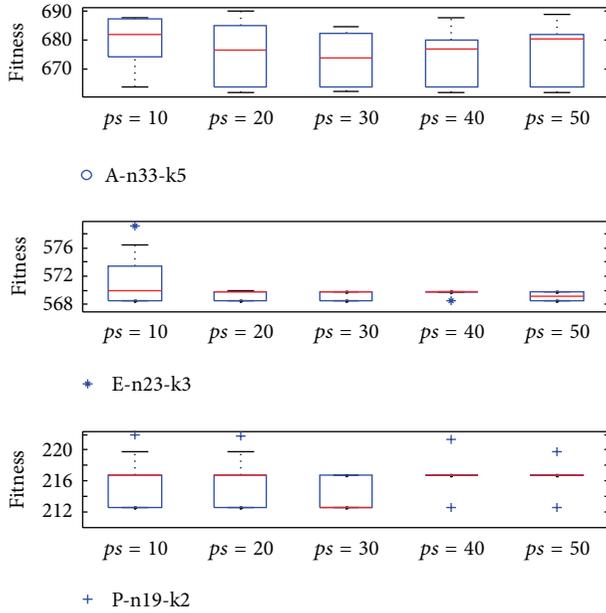
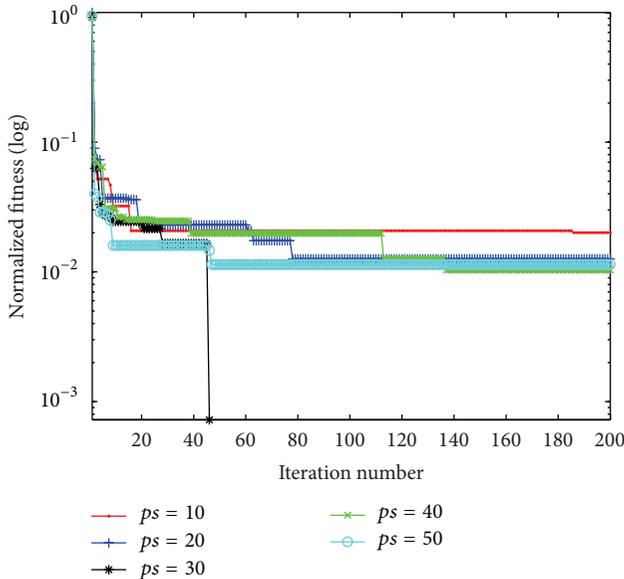
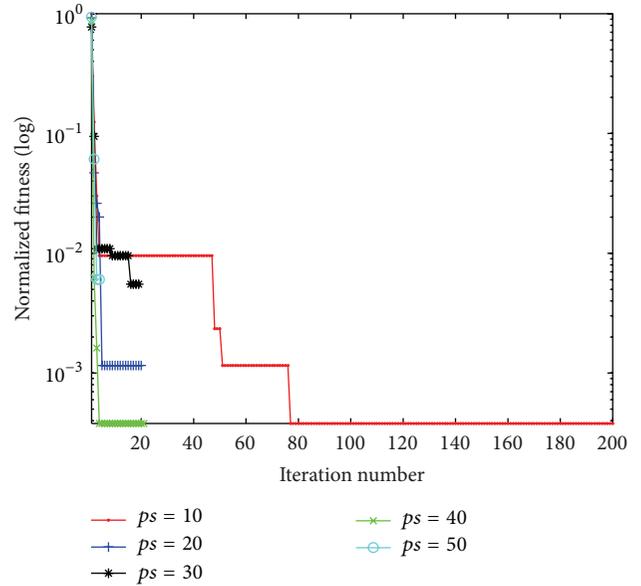
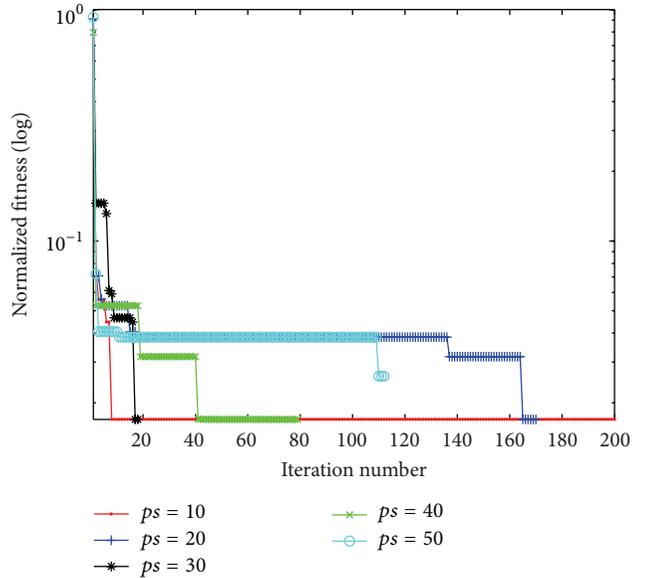


FIGURE 7: Box-and-whisker diagram of selected three instances.

FIGURE 8: Convergence curve for *A\_n33\_k5*.

convergence, the parameter  $ps$  takes a compromise values,  $ps = 30$ .

**4.2. Comparisons of Simulation Results.** In order to show the effective of HBA-PR, we carry out a simulation to compare HBA-PR with other state-of-art algorithms, that is, a parallel version of the classical Clarke and Wright Savings (CWS) heuristic and SR-GCWS (Simulation in Routing via the Generalized Clarke and Wright Savings heuristic) proposed by Juan et al. [9], CS-GRASP proposed by Zheng et al. [10]. Results of these simulations are summarized in Table 1, which contain the following information of each instance: vehicle

FIGURE 9: Convergence curve for *E\_n23\_k3*.FIGURE 10: Convergence curve for *P\_n19\_k2*.

capacity; tightness (demand/capacity); I.BKS is integral best-known solution (BKS) or “optimal” value according to the web <http://www.branchandcut.org/>; R\_BSK is verifies real costs for the best-known solution according to [9]; CWS is the costs associated with the solution given by the parallel version of the CWS heuristic; SR-GCWS is the best solution obtained by SR-GCWS method; HBA-PR is our best solution, where “—” represents no records in the literature.

From the simulation results obtained by testing HBA-PR, it demonstrates that using the proposed HBA-PR to solve the CVRP is effective, and the performance of HBA-PR is prominent. From Table 1, all instances achieved a good quality

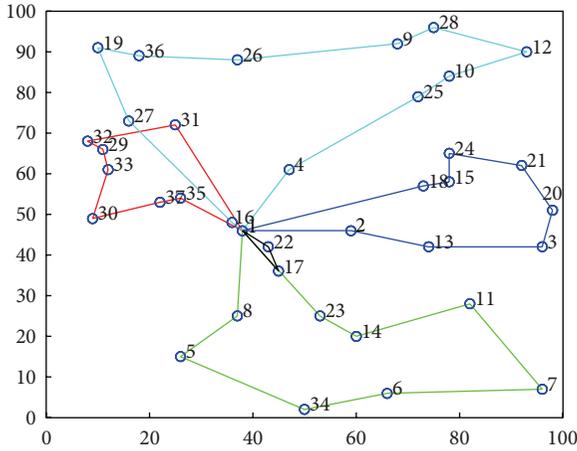


FIGURE 11: Optimal routes of  $A_{n37\_k5}$ .

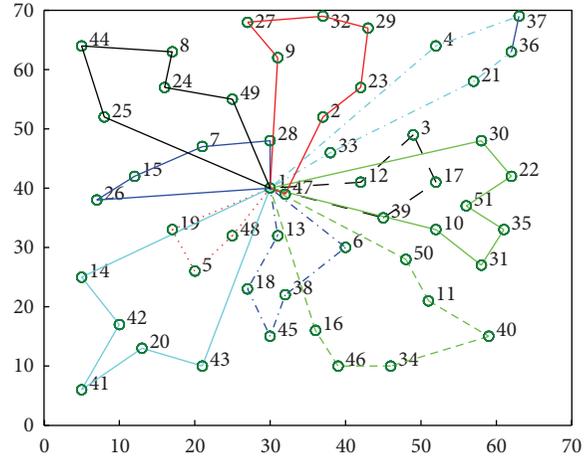


FIGURE 13: Optimal routes of  $P_{n51\_k10}$ .

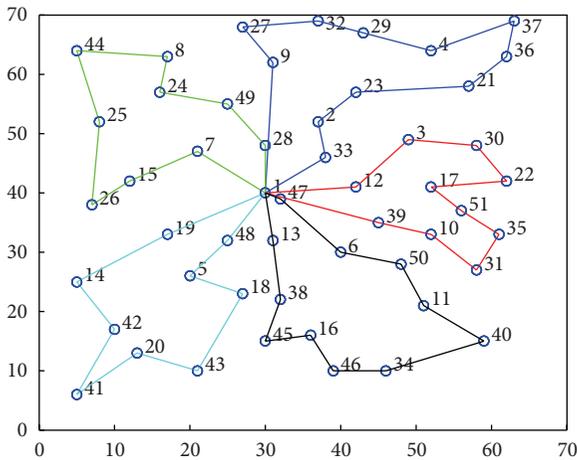


FIGURE 12: Optimal routes of  $E_{n51\_k5}$ .

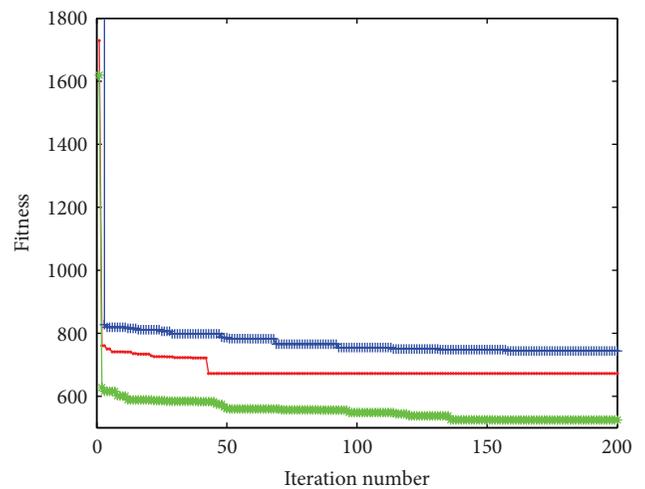


FIGURE 14: Convergence curve of three instances.

solution, and solutions of 6 instances are better than the best-known solution (“optimal” value). HBA-PR has matched 10 of the 12 best-known solutions except for  $P_{n51\_k10}$  and  $A_{n39\_k6}$ , and the average deviation from the real costs best-known solution is 0.057%. HBA-PR outperforms CWS for the 12 instances. Comparing with CS-GRASP and HBA-PR, HBA-PR outperforms CS-GRASP in terms of  $A_{n33\_k5}$ ,  $A_{n33\_k6}$ , and  $E_{n23\_k3}$ ; the other instances have same results. The SR-GCWS and HBA-PR have similar results, and the gap is very small.

Furthermore, Figure 11 shows the best solution found so far by using our methodology for the  $A_{n37\_k5}$ .vrp file, where the depot (using 1 instead of 0) is at the center. Analogously, Figures 12 and 13 show the best solution found by HBA-PR for the  $E_{n51\_k5}$ .vrp file and  $P_{n51\_k10}$ .vrp file. The convergence curve for some test problems has been shown in Figure 14. From Figure 14, for instance,  $A_{n37\_k5}$ , it converges to an optimal solution after 43 generation, and it expends about 170 generations, for instance,  $E_{n51\_k5}$  and  $P_{n51\_k10}$  when algorithm converges to an optimal solution, which demonstrate that the HBA-PR has a faster convergence rate.

In general, the proposed HBA-PR can produce good solutions when compared with existing heuristics for solving the CVRP, and the convergence rate of HBA-PR is faster. These results seem to indicate that the hybrid bat algorithm with path relinking is an alternative to solve the capacitated vehicle routing problem.

### 5. Conclusions

The capacitated vehicle routing problem is important in the fields of Operations Research, which is an NP-hard problem. Bat algorithm is a continuous metaheuristics; it cannot be used to solve the CVRP directly. In this paper, a hybrid bat algorithm with path relinking (HBA-PR) for solving the CVRP has been presented. This methodology, which does not require any particular fine tuning of parameters or configuration process, combines the classical greedy randomized

adaptive search procedure (GRASP) with bat algorithm; the path relinking as an intensification strategy to explore local trajectories connecting elite solutions obtained by proposed algorithm; subsequence and single-point local search are effectively integrated into HBA-PR. Results show that our methodology is able to provide fine-quality solutions which can compete with the ones provided by some exact and heuristic approaches. Moreover, because of its simplicity and flexibility, we think that this methodology can easily be adapted to other variants of the vehicle routing problem and even to other combinatorial problems, for example, the vehicle routing problem with time window and traveling salesman problem, which is our further work.

## Acknowledgments

This work is supported by National Science Foundation of China under Grant no. 61165015, Key Project of Guangxi Science Foundation under Grant no. 2012GXNSFDA053028, Key Project of Guangxi High School Science Foundation under Grant no. 20121ZD008, the Funded by Open Research Fund Program of Key Lab of Intelligent Perception and Image Understanding of Ministry of Education of China under Grant no. IPIU01201100, and the Innovation Project of Guangxi Graduate Education under Grant no. YCSZ2012063.

## References

- [1] G. B. Dantzig and J. H. Ramser, "The truck dispatching problem," *Management Science*, vol. 6, pp. 80–91, 1959.
- [2] M. Haimovich, A. H. G. Rinnooy Kan, and L. Stougie, "Analysis of heuristics for vehicle routing problems," in *Vehicle Routing: Methods and Studies*, B. L. Golden and A. A. Assad, Eds., vol. 16, pp. 47–61, North-Holland, Amsterdam, The Netherlands, 1988.
- [3] P. Toth and A. Tramontani, "An integer linear programming local search for capacitated vehicle routing problems," in *The Vehicle Routing Problem: Latest Advances and New Challenges*, pp. 275–295, Springer, New York, NY, USA, 2008.
- [4] H. Nazif and L. S. Lee, "Optimised crossover genetic algorithm for capacitated vehicle routing problem," *Applied Mathematical Modelling*, vol. 36, no. 5, pp. 2110–2117, 2012.
- [5] T. J. Ai and V. Kachitvichyanukul, "Particle swarm optimization and two solution representations for solving the capacitated vehicle routing problem," *Computers and Industrial Engineering*, vol. 56, no. 1, pp. 380–387, 2009.
- [6] W. Y. Szeto, Y. Wu, and S. C. Ho, "An artificial bee colony algorithm for the capacitated vehicle routing problem," *European Journal of Operational Research*, vol. 215, no. 1, pp. 126–135, 2011.
- [7] P. Chen, H.-K. Huang, and X.-Y. Dong, "Iterated variable neighborhood descent algorithm for the capacitated vehicle routing problem," *Expert Systems with Applications*, vol. 37, no. 2, pp. 1620–1627, 2010.
- [8] A. Yurtkuran and E. Emel, "A new hybrid electromagnetism-like algorithm for capacitated vehicle routing problems," *Expert Systems with Applications*, vol. 37, no. 4, pp. 3427–3433, 2010.
- [9] A. A. Juan, J. Faulin, R. Ruiz, B. Barrios, and S. Caballé, "The SR-GCWS hybrid algorithm for solving the capacitated vehicle routing problem," *Applied Soft Computing Journal*, vol. 10, no. 1, pp. 215–224, 2010.
- [10] H. Q. Zheng, Y. Q. Zhou, and Q. F. Luo, "A hybrid cuckoo search algorithm-GRASP for vehicle routing problem," *Journal of Convergence Information Technology*, vol. 8, no. 3, 2013.
- [11] X.-S. Yang, "A new metaheuristic Bat-inspired algorithm. Nature inspired cooperative strategies for optimization (NICSO)," *Studies in Computational Intelligence*, vol. 284, pp. 65–74, 2010.
- [12] A. H. Gandom, X. S. Yang, A. H. Alavi, and S. Talatahari, "Bat algorithm for constrained optimization tasks," *Neural Computing and Applications*, vol. 22, no. 6, pp. 1239–1255, 2013.
- [13] X. S. Yang and A. H. Gandom, "Bat algorithm: a novel approach for global engineering optimization," *Engineering Computations*, vol. 29, pp. 464–483, 2012.
- [14] S. Mishra, K. Shaw, and D. Mishra, "A new meta-heuristic bat inspired classification approach for microarray data," *Procedia Technology*, vol. 4, pp. 802–806, 2012.
- [15] J. Xie, Y. Q. Zhou, and H. Chen, "A novel bat algorithm based on differential operator and Lévy-flights trajectory," *Computational Intelligence and Neuroscience*, vol. 2013, Article ID 453812, 13 pages, 2013.
- [16] G. Wang, L. Guo, H. Duan, L. Liu, and H. Wang, "A bat algorithm with mutation for UCAV path planning," *The Scientific World Journal*, vol. 2012, Article ID 418946, 15 pages, 2012.
- [17] T. A. Feo and M. G. C. Resende, "Greedy randomized adaptive search procedures," *Journal of Global Optimization*, vol. 6, no. 2, pp. 109–133, 1995.
- [18] M. G. C. Resendel and C. C. Ribeiro, "GRASP with path-relinking: recent advances and applications," in *Metaheuristics: Progress as Real Problem Solvers*, pp. 29–63, Springer, 2005.
- [19] P. Festa and M. G. C. Resende, "An annotated bibliography of GRASP. I. Algorithms," *International Transactions in Operational Research*, vol. 16, no. 1, pp. 1–24, 2009.
- [20] P. Festa and M. G. C. Resende, "An annotated bibliography of GRASP. II. Applications," *International Transactions in Operational Research*, vol. 16, no. 2, pp. 131–172, 2009.
- [21] M. Mestria, L. S. Ochi, and S. de Lima Martins, "GRASP with path relinking for the symmetric euclidean clustered traveling salesman problem," *Computers and Operations Research*, 2012.
- [22] M. G. C. Resende, C. C. Ribeiro, F. Glover, and R. Mart, "Scatter search and path-relinking: fundamentals, advances, and applications," in *Handbook of Metaheuristics*, M. Gendreau and J. Y. Potvin, Eds., pp. 87–107, Springer, Boston, Mass, USA, 2010.
- [23] F. Glover, "Tabu search and adaptive memory programming—advances, applications and challenges," in *Interfaces in Computer Science and Operations Research*, R. S. Barr, R. V. Helgason, and J. L. Kennington, Eds., pp. 1–75, Kluwer, 1996.
- [24] M. Laguna and R. Martí, "GRASP and path relinking for 2-layer straight line crossing minimization," *Journal on Computing*, vol. 11, no. 1, pp. 44–52, 1999.

## Research Article

# Ripple-Spreading Network Model Optimization by Genetic Algorithm

Xiao-Bing Hu,<sup>1,2</sup> Ming Wang,<sup>1</sup> and Mark S. Leeson<sup>2</sup>

<sup>1</sup> State Key Laboratory of Earth Surface Processes and Resource Ecology, Beijing Normal University, 100875 Beijing, China

<sup>2</sup> School of Engineering, University of Warwick, Coventry CV4 7AL, UK

Correspondence should be addressed to Xiao-Bing Hu; [dr\\_xiaobinghu@hotmail.co.uk](mailto:dr_xiaobinghu@hotmail.co.uk)

Received 3 May 2013; Accepted 19 July 2013

Academic Editor: Vishal Bhatnagar

Copyright © 2013 Xiao-Bing Hu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Small-world and scale-free properties are widely acknowledged in many real-world complex network systems, and many network models have been developed to capture these network properties. The ripple-spreading network model (RSNM) is a newly reported complex network model, which is inspired by the natural ripple-spreading phenomenon on clam water surface. The RSNM exhibits good potential for describing both spatial and temporal features in the development of many real-world networks where the influence of a few local events spreads out through nodes and then largely determines the final network topology. However, the relationships between ripple-spreading related parameters (RSRPs) of RSNM and small-world and scale-free topologies are not as obvious or straightforward as in many other network models. This paper attempts to apply genetic algorithm (GA) to tune the values of RSRPs, so that the RSNM may generate these two most important network topologies. The study demonstrates that, once RSRPs are properly tuned by GA, the RSNM is capable of generating both network topologies and therefore has a great flexibility to study many real-world complex network systems.

## 1. Introduction

As is well known, many successful artificial intelligence technologies are actually inspired by certain natural systems or phenomena [1–3]. For instance, genetic algorithms (GAs) are inspired by natural selection and evolutionary processes, artificial neural networks (ANNs) by the animal brain, particle swarm optimization (PSO) by the learning behavior within a population, ant colony optimization (ACO) by the foraging behavior of ants, and optimal structure design by the hexagonal cells of the honeycomb.

Following the common practice of learning from nature in the artificial intelligence domain, we have recently proposed a ripple-spreading network model (RSNM) [4], which is inspired by the natural ripple-spreading phenomenon. A ripple-spreading process defined by some ripple-spreading related parameters (RSRPs) is the central piece of the model. Basically, assuming that nodes of networks are distributed in a (real or artificial) space and that the distribution of nodes are fixed, some initial ripples are randomly generated in the space. As an initial ripple spreads out in the space, its point

energy decays gradually, and it reaches every node one by one sooner or later. By comparing the point energy of an incoming ripple with some preset thresholds for a node, it can be determined whether this node will be activated by the incoming ripple to generate a new ripple, and whether this node will be connected to the node where the incoming ripple originates. When a node is activated to generate a new ripple, the initial energy of the new ripple will be a function of the point energy of the incoming ripple. New ripples are also able not only to activate other nodes to generate more ripples but also to establish new connections between nodes, as long as their point energy is above the relevant thresholds. As this ripple-spreading process goes on for a while, a network topology will appear. Actually, the output topology will be largely determined by the values of RSRPs, such as the locations of the epicenters for initial ripples, the thresholds to tell whether a node will be activated or connected, the energy amplifying factor, and the coefficients to define the point energy decaying rate. Once the values for these RSRPs are given and fixed, then the output topology will be uniquely determined.

However, when compared with many existing network models such as those reported in [5, 6], there is a lack of obvious or straightforward relationships between the RSRPs and the network properties of RSNM topology. In particular, [4] listed it as one of the future works to study when and how the RSNM may generate two of the most important network topologies, small-world and scale-free topologies. Therefore, based on the theoretical and conceptual results in [4], this paper focuses in particular on providing a practicable method to tune the RSRPs of RSNM, in order to generate preferable network topologies. To this end, the model in [4] will be modified into a more general form to ease the analysis on the properties of RSNM, and then an effective GA will be developed, in order to tune the RSNM to generate small-world and scale-free network topologies.

The remainder of this paper is organized as follows. Section 2 gives some background knowledge of related work. Section 3 describes a mathematical model of the modified RSNM. The proposed GA will be explained in Section 4 for optimizing the values of RSRPs. Section 5 reports some simulation results. The paper ends with some conclusions and discussions on future work in Section 6.

## 2. Related Work on Complex Networks

Complex networks, that is, networks whose structure is irregular, complex, and dynamically evolving in time, describe a wide range of systems in nature and society and are all around us in our daily life [6, 7]. In the past few decades, many efforts have been made to model and analyze various complex networks. Most network models can be classified as stochastic models because they have a typical feature in common: a stochastic model abstracts one or a few network properties or parameters to capture in quantitative terms the underlying organizing principles of complex networks, and these network properties or parameters can estimate how the network appears but cannot guarantee an exact or unique topology. In other words, different network topologies may have exactly the same values for the specified properties or parameters. For instance, in classic random graph theory [5], the connection probability is the core parameter of the network model. In the generalized random graph model [8], a degree distribution following a power law is used as the input in order to be able to describe the scale-free character of real networks that the classic random graph model cannot capture. In the theory of evolving networks [9], the parameter known as the preferential attachment is often used to model scale-free real networks. In a recently reported spatially embedded random network model [10], the connection probability is formulated as a function of distance between nodes. In either of the above network models, even if the input is fixed, for example, the connection probability, power law for degree distribution or preferential attachment is fixed, the output of the model is enormous, and what the output topology exactly looks like is largely by chance. Therefore, they are all stochastic models. With the exception of those network properties which explicitly depend on the input parameters, the output topology of the models is

largely unpredictable or uncertain in terms of other network properties. As a result, it is difficult to apply these models in network design, such as the topology optimization problem.

In contrast, the RSNM proposed in [4] is a deterministic complex network model. Unlike the stochastic models in [5, 8–10], once the values of input are fixed, the output network topology will also be fixed and unique in the new model. Therefore, we can adjust these RSRPs in order to improve the network topology in terms of concerned network properties. Furthermore, there is great freedom and flexibility to modify and extend the RSNM. For instance, the proposed RSNM can easily be extended to a semideterministic version and a stochastic version. Basically, the RSNM can introduce randomness at two levels by RSRPs: (i) the random setup of nodes (e.g., the random spatial distribution of nodes and the random distribution of node thresholds and amplifying factors, if they are not constants); (ii) the random setup of initial stimulating ripples (e.g., locations, initial energy, and starting time). The first level of randomness is related to the internal factors of the system, and the second level of randomness is associated with external impacts to the system. The two levels of randomness in the RSNM make it very easy and natural for the RSNM to capture/reflect both internal and external factors in many real-world complex networks. For instance, the impact of earthquake through a risk chain and the breakout of plagues in a community can be described as evolving networks triggered by a few initial simulating ripples; the threshold of nodes is related to the vulnerability of infrastructure and the immunity of individuals and the amplifying factor of nodes to connectivity of infrastructure and the social activeness of individuals.

Any complex network model claimed to be practicable must illustrate the capability of generating network topologies that present in real-world systems, among which small-world and scale-free topologies are two of the most important network topologies. It is widely recognized that many real-world complex network systems display some organizing principles, which often lead to either small-world topology or scale-free topology [7, 11, 12]. The small-world concept in simple terms describes the fact that despite their often large size, in most networks, there is a relatively short path between any two nodes. The distance between two nodes is defined as the number of edges along the shortest path connecting them. The most popular manifestation of small worlds is the “six degrees of separation” concept, which concluded that there was a path of acquaintance with a typical length of about six between most pairs of people in the United States [13]. The small-world property appears to characterize most complex networks, for instance, the actors in Hollywood are on average within three costars from each other or the chemicals in a cell are typically separated by three reactions. Another most important development in our understanding of complex networks was the discovery that for a large number of networks, including the World Wide Web [14], the Internet [15], or metabolic networks [16], the degree distribution has a power-law tail. Such networks are called scale free [17]. While some networks display an exponential tail, often the functional form of degree distribution still deviates significantly from the Poisson distribution expected

for a random graph. The discoveries of small-world and scale-free features have fundamentally reshaped the study of complex network systems. Therefore, to be able to better apply the RSNM in [4] to study real-world complex networks, it is crucial to understand how and when the RSNM will generate small-world and scale-free network topologies, which is the focus of this paper.

### 3. Mathematical Description of Ripple-Spreading Network Model (RSNM)

**3.1. The Basic Idea of RSNM.** The RSNM was first reported in [4], where the basic idea was inspired by the natural ripple-spreading phenomenon on calm water surfaces. Suppose that a collection of stakes is randomly distributed in a quiet pond. Then, suddenly a stone is thrown into the pond, and an initial ripple is generated and spreads out from the point where the stone hits the quiet water. When the ripple reaches a nearby stake, a new ripple is generated around the stake due to the reflection effect. For the sake of consistency, we call such a new ripple a *responding* ripple (or outgoing ripple), and the ripple which triggers the responding ripple as a *stimulating* ripple (or incoming ripple). Obviously, the initial energy of a responding ripple is determined by both the point energy of the stimulating ripple and the physical features of the stake (e.g., whether the stake is rigid or soft in material). A responding ripple may trigger new responding ripples around other stakes. As the initial stimulating ripple spreads out, more and more responding ripples are stimulated around stakes. However, since the point energy on the stimulating ripples decays as they spread out, those responding ripples triggered at a late phase could hardly be noticed. Therefore, we may say there is a threshold, and only when the point energy of a stimulating ripple is above this threshold will a responding ripple be generated. Furthermore, the reaction of stake to a stimulating ripple may be more than just generating a responding ripple. For instance, if the point energy of a stimulating ripple is above another threshold, the stake will shake noticeably to disturb a bird resting on the top of the stake. Once the bird on a stake is disturbed by the outgoing ripple of another stake, we set up an edge between the associated pair of nodes in a network. As the time elapses, the point energy of all ripples will decay to below the thresholds sooner or later, and after a long enough period of time, all ripples will disappear and no birds will be disturbed any more. Then, based on the established edges according to which bird has been disturbed by the outgoing ripple of which stake, we will get a network topology. Clearly there are some factors affecting the final network topology. For instance, how many stones hit the pond to generate initial stimulating ripples, where do they hit the pond, what is the mass of each stone (will determine the initial energy of the associated stimulating ripple), what are the preset values for the thresholds, and what is the amplifying factor of each stake to reflect a stimulating ripple? By mathematically formulating these factors and the relationships between them, we can get an effective model for complex networks. For more details of the basic idea of RSNM, readers may refer to [4].

**3.2. Mathematical Model of Modified RSNM.** The RSNMs proposed in [4] have an important concept: epicenter of initial stimulating ripple (EISR), which is associated with the point where the stone hit the water surface. Unlike normal nodes in network, an EISR is not connectable, which means no edge can be established between two EISRs or an EISR and a normal node. EISRs are useful to match and describe the initial cause of the development of a real-world network; for instance, in the study on the impact of earthquake on a transport network, the epicenter of earthquake is usually not located at any node of the transport network. However, the difference between EISRs and normal nodes makes it very difficult to conduct theoretical deduction of network properties. This partially explains why paper [4] failed to analyze the small-world and scale-free properties of the RSNM. Without necessary theoretical analyses, the RSNM can hardly develop into a mature and practicable method. From a simulation point of view, a network model with EISRs can be mathematically equivalent to a model without EISRs, and the latter will be much easier to analyze and study. Therefore, in this paper, we will modify the models in [4] by removing EISRs and adding a new concept of self-generating ripple (SGR). Basically, in the modified RSNM all initial stimulating ripples will be self-generated by connectable nodes in the network, in other words, all initial stimulating ripples will directly come from connectable nodes. To this end, two new ripple-spreading related parameters (RSRPs) need to be introduced to every node: initial energy of SGR  $E_{\text{SGR}}(i)$  and start time of SGR  $T_{\text{SGR}}(i)$ ,  $i = 1, \dots, N_N$ , where  $N_N$  is the total number of nodes in the network. By presetting  $E_{\text{SGR}}(i)$  and  $T_{\text{SGR}}(i)$  according to the impact of EISRs, one can easily see that the modified RSNM is equivalent to the models reported in [4]. Actually, the modified RSNM is a generalized model, as the models in [4] are just a subset of the new model. Obviously, only those nodes with  $E_{\text{SGR}}(i) > 0$  matter in terms of EISRs, and we assume that there are  $N_{\text{SGR}} \leq N_N$  such nodes.

Before we give the mathematical description of the modified RSNM, we need to define some other important ripple-spreading related parameters (RSRPs) like in [4]. For the sake of simplicity but without loss of generality, unless explicitly specified, we suppose all nodes are distributed in a limited 2-dimensional space in this paper. It is also assumed that the total number of nodes,  $N_N$ , and their locations,  $(x_N(i), y_N(i))$ , are already given and fixed. To get different topologies from this fixed set of nodes, we have another three RSRPs to each node:  $\alpha(i)$ ,  $\beta_R(i)$ , and  $\beta_L(i)$ , which are the amplifying factor, the threshold to generate a responding ripple, and the threshold to establish a link, for node  $i$ ,  $i = 1, \dots, N_N$ , respectively.

With the above RSRPs, the modeling process of the modified RSNM can be mathematically described as follows.

*Step 1.* Initialize the current time instant, that is,  $t = 0$ . Initialize  $E_{\text{SGR}}(i)$  and  $T_{\text{SGR}}(i)$  for every node randomly or according to certain requirements. Since each node has no initial energy, that is,  $E_N(i) = 0$ , therefore, its current point energy is

$$e_N(i, t) = E_N(i) = 0, \quad i = 1, \dots, N_N. \quad (1)$$

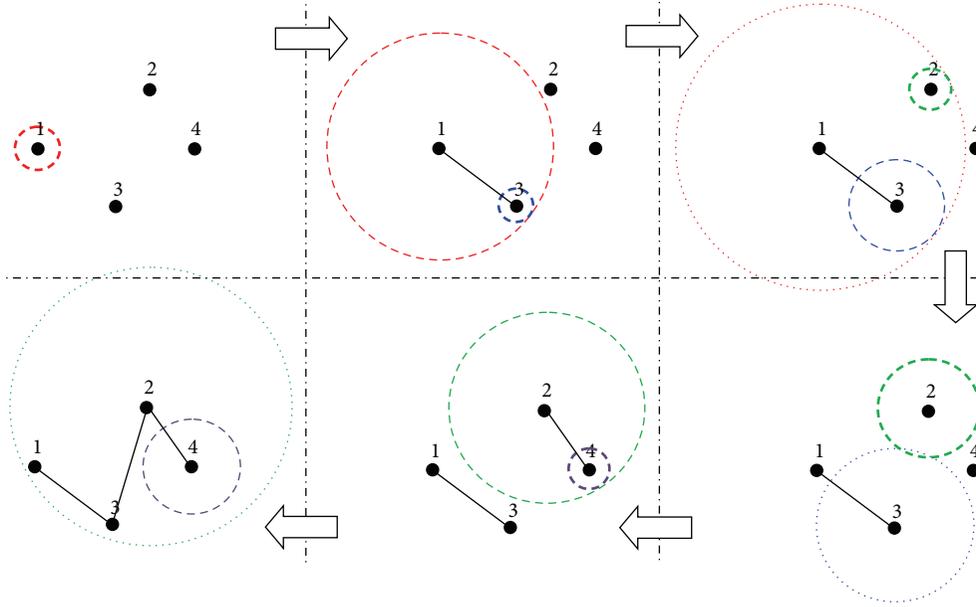


FIGURE 1: The development of a simple network topology based on the RSNM, where the epicenter of initial stimulating ripple (EISR) is a connectable node in the network, that is, node 1.

Assume that each node has a ripple with a current radius of 0, that is,  $r_N(i, t) = 0$ .

*Step 2.* If the termination criteria are not satisfied, do as follows.

*Substep 1.* Let  $t = t + 1$ .

*Substep 2.* Check  $t$  against  $T_{SGR}(i)$ . If  $t \geq T_{SGR}(i)$ ,  $E_{SGR}(i) > 0$ , and  $r_N(i, t) = 0$ , then the self-generated ripple of node  $i$  will start to spread, so set up the initial energy of the ripple starting from node  $i$  as

$$E_N(i) = E_{SGR}(i). \quad (2)$$

*Substep 3.* If  $E_N(i) > 0$ ,  $i = 1, \dots, N_N$ , then update the current radius and point energy of the ripple starting from node  $i$

$$\begin{aligned} r_N(i, t) &= r_N(i, t-1) + s, \\ e_N(i, t) &= f_{Decay}(E_N(i), r_N(i, t), t), \end{aligned} \quad (3)$$

where  $s$  is the spreading speed of ripples, that is, the change in the radius of a ripple during one time instant, and  $f_{Decay}$  is a function defining how the point energy decays as the ripple spreads out. A typical decaying function may be

$$f_{Decay}(E_N(i), r_N(i, t), t) = \eta \frac{E_N(i)}{2\pi r_N(i, t)}, \quad (4)$$

where  $\eta$  is a coefficient and  $\pi$  is the mathematical constant. Clearly  $\eta$  has an important influence on the decaying speed of ripples and will therefore affect the final network topology. Basically, (3) show that the ripples have the same spreading speed and the same decaying function, just like in the natural

world. However, this may be amendable in order to get a more complicated artificial model.

*Substep 4.* Check which new nodes are reached by the ripples. Suppose that  $L_N(i, j)$  is the distance between node  $i$  and node  $j$ . If  $E_N(j) = 0$  and  $L_N(i, j) \leq r_N(i, t)$ , then node  $j$  is reached by the ripple generated by node  $i$ . If  $e_N(i, t) \geq \beta_R(j)$ , then node  $j$  is activated by node  $i$  and generates a responding ripple with

$$E_N(j) = \begin{cases} \alpha(j) e_N(i, t), & E_{SGR}(j) \leq \alpha(j) e_N(i, t), \\ E_{SGR}(j), & E_{SGR}(j) > \alpha(j) e_N(i, t). \end{cases} \quad (5)$$

If  $e_N(i, t) \geq \beta_L(j)$ , then a connection between node  $i$  and node  $j$  is established, that is,

$$A(i, j) = A(j, i) = 1, \quad (6)$$

where  $A$  is the adjacency matrix which records the network topology.

Different termination criteria may be used in Step 2. For instance, the current time instance is beyond a specific time window, no node has current point energy above any threshold, or the upper bound for the number of total links is reached.

Figure 1 gives a simple illustration of the development of network topology based on the modified RSNM, where the initial stimulating ripple directly comes from a connectable node in the network. From Figure 1, one may notice that since the threshold for triggering a responding ripple may differ from the threshold for establishing an edge, a responding ripple is not always accompanied by the establishment of an edge. For instance, the outgoing ripple of node 1 triggers

a responding ripple at node 3 accompanied by the establishment of an edge to node 2, it only triggers a responding ripple at node 2, and it does nothing to node 4.

**3.3. Some Important Conditions.** We have previously presented some conditions for analyzing general network properties of the EISR-based RSNM [4], and here, we need to work out new conditions for the modified RSNM based on the concept of SGR. The proofs of new conditions are similar to those in [4], to which readers can refer to for details. Such new conditions will be very useful as heuristic rules in the application of GA to optimize the output network topology of RSNM.

For a network with  $N_N$  nodes, the number of total potential edges between nodes is

$$N_{PE} = \frac{N_N \times (N_N - 1)}{2}. \quad (7)$$

For each pair of nodes  $(i, j)$ , let  $L_N(i, j)$  denote the length of the potential edge between node  $i$  and node  $j$ , that is, the direct distance between node  $i$  and node  $j$ , no matter whether or not there is an actual connection between the pair of nodes. Since all nodes are distributed in a limited space, we can assume that  $L_U$  and  $L_L$  are the maximal and minimal lengths in all potential edges, respectively,  $E_{SGR,U}$  and  $E_{SGR,L}$  the maximal and minimal values of all initial SGR energy  $E_{SGR}(i)$ ,  $\beta_{R,U}$  and  $\beta_{R,L}$  the maximal and minimal values of all activating threshold  $\beta_R(i)$ ,  $\beta_{L,U}$  and  $\beta_{L,L}$  the maximal and minimal values of all connecting threshold  $\beta_L(i)$ , and  $\alpha_U$  and  $\alpha_L$  maximal and minimal values of all amplifying factor  $\alpha(i)$ ,  $i = 1, \dots, N_N$ .

**Condition 1** (completely connected graph). If the inequalities

$$\eta \frac{E_{SGR,U}}{2\pi L_{\text{MaxMin}}} \geq \beta_{R,U}, \quad (8)$$

$$\eta \frac{\alpha_L \beta_{R,L}}{2\pi L_U} \geq \max(\beta_{R,U}, \beta_{L,U}) \quad (9)$$

hold, where

$$L_{\text{MaxMin}} = \max_{i=1, \dots, N_N} \min_{j=1, \dots, N_N} L_N(i, j), \quad (10)$$

then the RSNM will generate a completely connected graph.

Equation (8) guarantees that the SGRs will trigger at least one responding ripple, and (9) guarantees that the responding ripple will cause a completely connected graph.

**Condition 2** (all nodes connected). If (8) and the inequality

$$\eta \frac{\alpha_L \beta_{R,L}}{2\pi L_{\text{TSP}}} \geq \max(\beta_{R,U}, \beta_{L,U}) \quad (11)$$

hold, where  $L_{\text{TSP}}$  denotes the maximum edge length in a shortest open travelling salesman problem (TSP) route, then the RSNM will generate a graph where all nodes are connected.

**Condition 3** (all nodes activated). If (8) and the inequality

$$\eta \frac{\alpha_L \beta_{R,L}}{2\pi L_{\text{TSP}}} \geq \beta_{R,U} \quad (12)$$

hold, then every node will generate an outgoing ripple under the RSNM.

The exact node degree can be counted as follows: for node  $i$ ,  $i = 1, \dots, N_N$ , its degree is

$$D_N(i) = \sum_{n=1}^{N_N} \sum_{m=n+1}^{N_N} A(n, m), \quad (13)$$

where  $A$  is the adjacency matrix given in (9) and its entries are determined by the ripple-spreading process described in Section 3.2. After the ripple-spreading process terminates, one has

$$A(n, m) = \begin{cases} 1, & \frac{\eta E_N(n)}{2\pi L_N(n, m)} \geq \beta_L(m) \\ & \text{or } \frac{\eta E_N(m)}{2\pi L_N(n, m)} \geq \beta_L(n), \\ 0, & \frac{\eta E_N(n)}{2\pi L_N(n, m)} < \beta_L(m), \\ & \frac{\eta E_N(m)}{2\pi L_N(n, m)} < \beta_L(n). \end{cases} \quad (14)$$

Please note that (13) and (14) are used to calculate the exact degree of each node after a network topology is generated by the RSNM. To roughly estimate node degree distribution from the RSRPs without even running the RSNM, we have the following proposition.

**Proposition 1** (degree distribution estimation). *Suppose that every node will generate an outgoing ripple; for example, Condition 3 is satisfied. For node  $i$ , let  $L_{DD}(i)$  be the value of*

$$L_{DD}(i) = \frac{\eta \alpha_L \beta_R(i)}{2\pi \beta_{L,U}}. \quad (15)$$

*Then, in the graph generated by the RSNM, the degree of node  $i$ ,  $D_N(i)$ , is not smaller than the number of nodes to which the direct distance  $L_N(i, j)$  is not larger than  $L_{DD}(i)$ .*

## 4. Genetic Algorithm (GA) to Optimize Network Topology

**4.1. Optimize Network Topology by Evolving Model Parameters.** The optimization of network topology has long been a challenging task [18], as it is widely acknowledged as an NP-hard problem. As large-scale parallel stochastic search and optimization algorithms, genetic algorithms (GAs), if properly designed, have the capability of producing high quality solutions to NP-hard problems in an acceptable period of time [19]. Actually, GAs have already been used to optimize some network structures, for example, the topology optimization of CCS7 network [20], MPLS network [21] and airline route networks [22]. However, in such studies

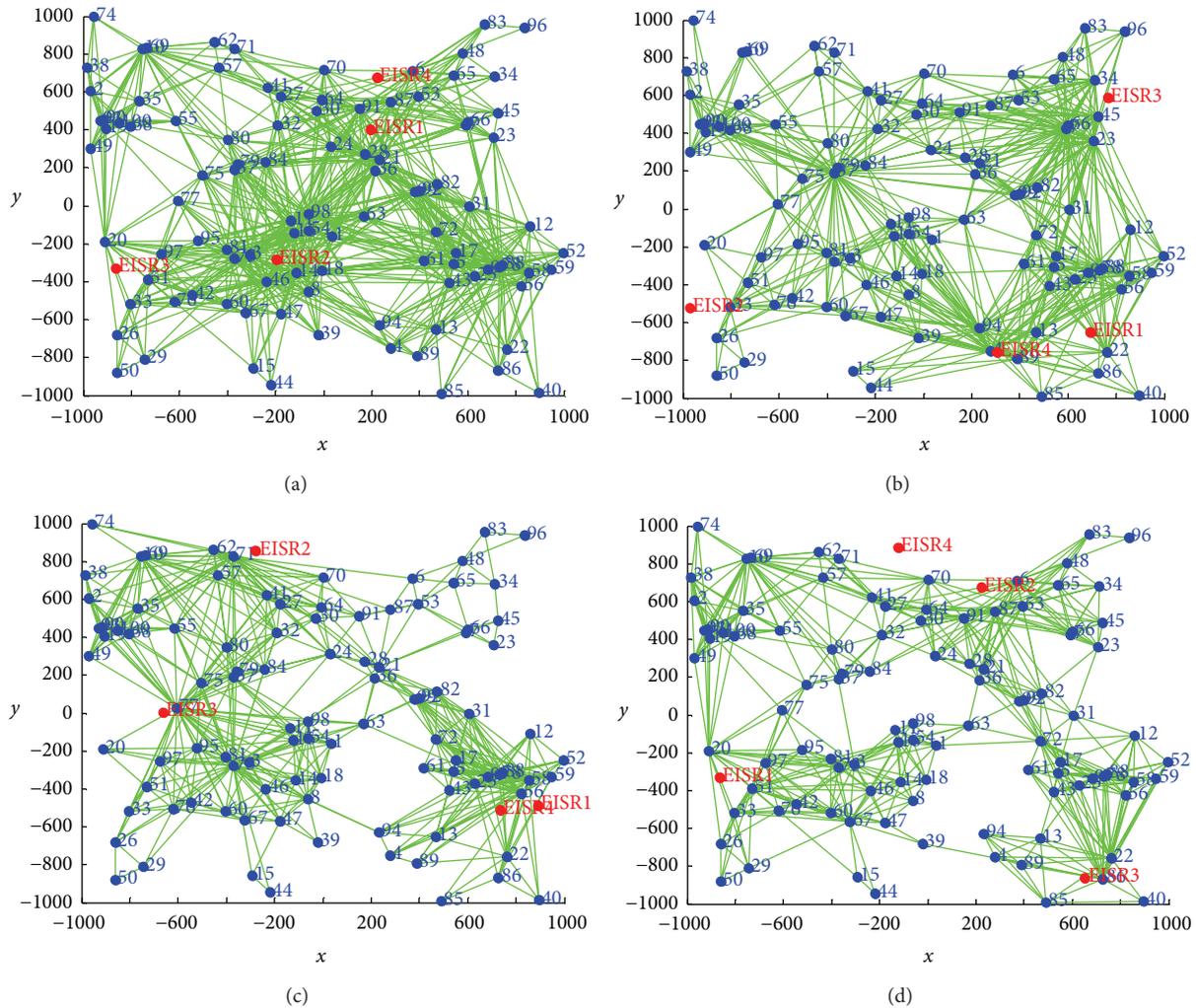


FIGURE 2: An illustration regarding how the distribution of EISRs affects network topology.

on network topology optimization, search is carried out by directly adding new links or removing existing links between nodes. The chromosome structure in such GAs is based on permutation representation of existing links. As is well known, permutation representation based GAs are often confronted with feasibility problems, which means random evolutionary operations, such as mutation and crossover, may generate some chromosomes whose associated solutions are invalid or infeasible against the underlying physical meaning of a real-world solution. For example, in the case of airline route network optimization, permutation representation based GA might establish a link between two very close cities, which is economically impracticable [22].

The RSNM makes it possible to discard permutation representation based GA, and to employ the very original binary representation based GA for network topology optimization. The original GA was developed with binary representation of value-based solutions (see [1, 2]), and it was free of feasibility problems. From the modeling process proposed in Section 3.2, one can see that the network topology is largely determined by the RSRPs. Actually, every given set of values for the RSRPs will result in a unique topology, and given

different values for the parameters, different topologies will be generated. Therefore, by evolving the values of RSRPs, one may find an optimal network topology in terms of particular considerations. Obviously, the very original binary representation based GA can be applied straightforwardly to evolve the RSRPs, just like in normal parameter optimization problems [2].

For the sake of simplicity, Figure 2 gives four examples about how the values of the RSRPs affect the output topology of the EISR-based RSNM. In the four networks in Figure 2, all other RSRPs have the same values, but the locations of the four EISRs are different. As a result, there are four different topologies. It should be pointed out that the feature that every given set of values for the model parameters will lead to a unique topology is distinct from other models of random networks, where the model parameters (e.g., the connection probability in the classic random graph theory and the preferential attachment in the theory of evolving networks) cannot uniquely determine network topology [5, 6].

One may argue that changing the values of RSRPs might never lead to every possible network topology. In other words, a certain topology might exist and no matter how

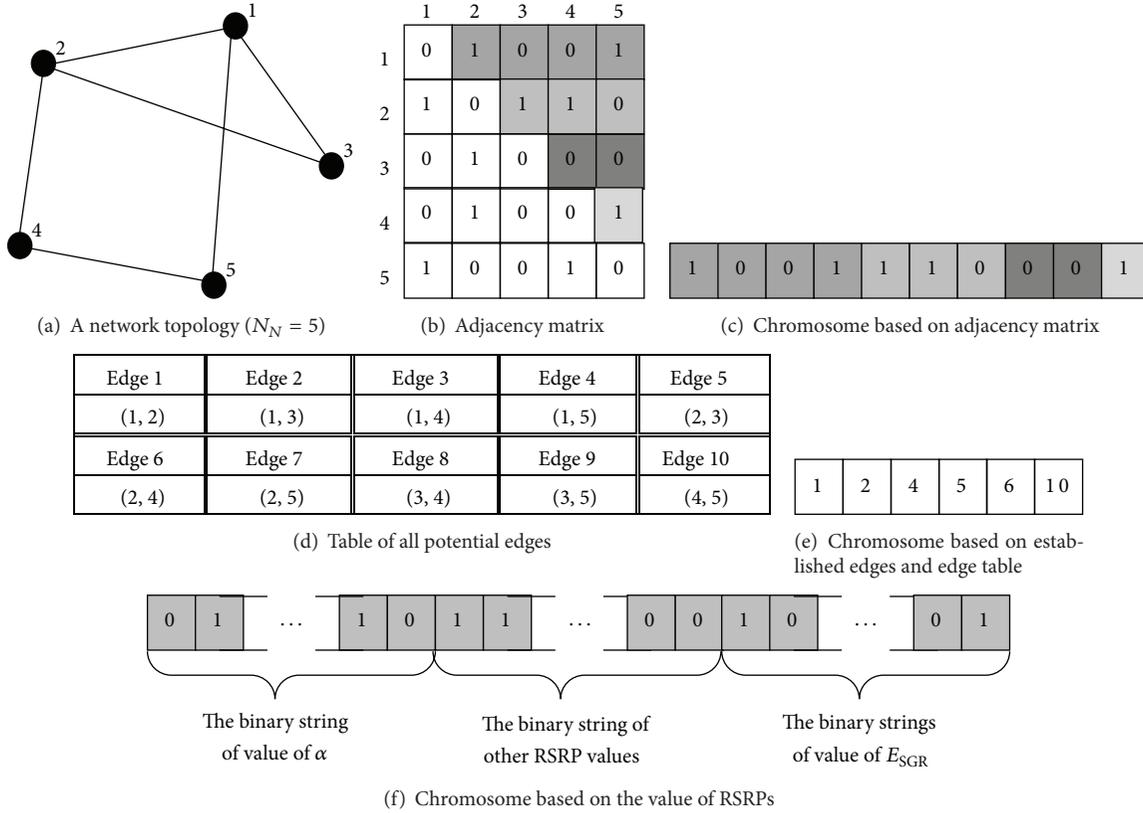


FIGURE 3: Some chromosome structures used in network topology optimization.

the RSRPs are tuned, the RSNM cannot generate it to the nature of the ripple-spreading process. Although this is true on the one hand, on the other, this feature of RSNM is not necessarily a disadvantage. In many applications of GAs, heuristic rules usually play a crucial role in delivering a good performance because they may filter out many infeasible or bad quality solutions and therefore narrow down the search space for GAs. The nature of the ripple-spreading process may do the same good thing for network topology optimization. Suppose that the development of a real-world network system largely involves some ripple-spreading effect (e.g., in the cases of disaster spreading and epidemic dynamics). Then, any network topology that cannot reflect such ripple-spreading effect is hardly realistic from the real-world system point of view and therefore should be considered as a bad solution to topology optimization. This means, if the ripple-spreading effect is concerned, then most network topologies missed out by the RSNM are possibly bad solutions. As demonstrated in [23], a ripple-spreading process can focus the search on quality solutions to the aircraft sequencing problem. Similarly, the RSNM can focus on more realistic topologies, and therefore, evolving the RSRPs could still be an effective strategy to find a good or even optimal network topology. Further study based on some particular real-world systems are needed, which is however beyond the scope of this paper.

**4.2. GA Design.** In the practice of GA design, we usually need to determine chromosome structure, mutation, crossover, and heuristic rules. These four design steps are often highly problem dependent in a permutation representation based GA but quite simple and straightforward in a binary representation based GA, which is the case in this study as just explained in Section 4.1.

Traditionally, to apply GA to optimize network topology, a straightforward chromosome structure is based on the adjacent matrix or a vector of established edges, just as illustrated in Figures 3(b)–3(e), and such a chromosome structure apparently belongs to permutation representation. Differently, with the RSNM, we can optimize network topology by evolving the values of RSRPs, and therefore, we can set the chromosome structure as a value based binary representation. Basically, the value of each RSRP can be represented as a binary string. We stick together all binary strings of a set of values for RSRPs, and then we get the associated chromosome (see Figure 3(f), e.g.). In the RSNM, a different node  $i$  may have different  $\alpha(i)$ ,  $\beta_R(i)$ , and  $\beta_L(i)$ , that is, different amplifying factors, threshold to generate a responding ripple, and threshold to establish a link, and a different SGR may have a different  $E_{SGR}(i)$ . In this study, we assume that  $\alpha(i)$  has a normal distribution within the range  $[\alpha_L, \alpha_U]$ ,  $\beta_R(i)$  within  $[\beta_{R,L}, \beta_{R,U}]$ ,  $\beta_L(i)$  within  $[\beta_{L,L}, \beta_{L,U}]$ , and  $E_{SGR}(i)$  within  $[E_{SGR,L}, E_{SGR,U}]$ . Therefore, we need to

use GA to tune the boundary values of the above 4 ranges, in other words, to use GA to optimize  $\alpha_L$ ,  $\alpha_U$ ,  $\beta_{R,L}$ ,  $\beta_{R,U}$ ,  $\beta_{L,L}$ ,  $\beta_{R,U}$ ,  $E_{SGR,L}$ , and  $E_{SGR,U}$ . As a result, a chromosome should be the integration of the binary strings of the above 8 boundary values. If we want to see what is the best topology, the RSNM could output with all nodes having the same value for certain RSRP, for example, demanding all nodes to have the same  $\alpha(i)$ , then a chromosome will become simpler because  $\alpha_L = \alpha_U$ . Furthermore, we may predetermine and fix certain RSRPs and use GA only to tune other RSRPs because, according to the conditions given in Section 3.3, one can see that it is not the absolute value of RSRPs, but the relative ratio between RSRPs that largely determine the output topology of the RSNM. This will further simplify the chromosome structure.

Mutation aims to increase the diversity of chromosomes, so that the GA can explore the solution space as widely as possible in order to stand a better chance to hit a global optimum. For a binary representation based GA, the mutation operator randomly selects some genes in a chromosome with a certain probability and reverses their values, that is, changing a binary bit from 1 to 0, or from 0 to 1.

Crossover is used to identify, inherit, and protect common genes shared by fit chromosomes, and at the same time, to recombine noncommon genes searching for new solutions. Crossover is crucial for GAs to quickly locate optima or suboptima. The simplest crossover is one-point crossover, where a split point is chosen randomly, each of the two parents split at the chosen point into two pieces, and piece 1 (or piece 2) from parent 1 is combined with piece 2 (or piece 1) from parent 2 to generate offspring. In this paper, we choose uniform crossover, mainly because it is more powerful in terms of exploiting all possibilities of recombining noncommon genes [24]. In uniform crossover, each gene in the offspring inherits the same gene from either parent 1 or parent 2 at a half-to-half chance. Actually, from Figure 4, one can see that uniform crossover is the ultimate multipoint crossover.

Problem-specific heuristic rules are used in GA to filter out bad/invalid/infeasible solutions and therefore to accelerate the converge speed. They are usually more useful and more important for permutation representation based GA rather than binary representation based GA. In the RSNM, although the ripple-spreading process is supposed to rule out most unrealistic solutions, we can still introduce some heuristic rules based on those conditions given in Section 3.3. For example, when initializing a chromosome, the random setup of RSRP values may more or less refer to those ratio relationships between the RSRPs, in order to avoid such cases where many nodes are not connected or all nodes are completely connected.

In the GA, the population of chromosomes evolves based on their fitness through generations of evolutionary operations: selection, mutation, and crossover. Since the goal in this study is to tune the RSNM to generate certain desirable topologies, the fitness of a chromosome is basically measured by network properties of the topology associated with the chromosome. It is known that small-world and scale-free topologies are mainly recognized by their node degree

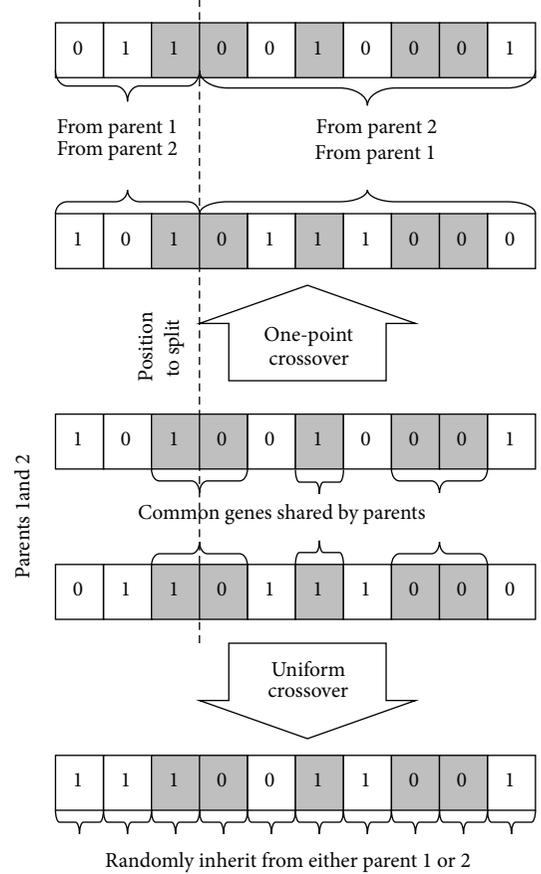
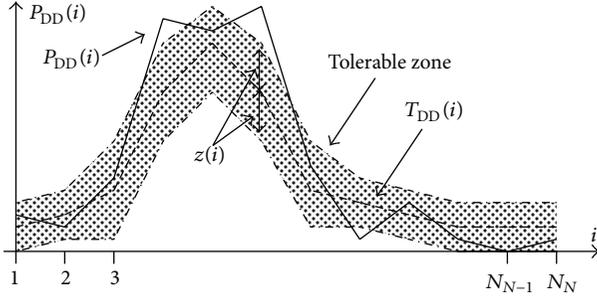


FIGURE 4: One-point crossover versus uniform crossover.

distribution pattern, average path length (APL) between connected nodes, and the clustering coefficient (CC). Basically, a small-world network topology has a Poisson distribution, small APL, and large CC, while a scale-free network topology has a power law tail-off distribution, even smaller APL, and larger CC. To apply a GA to tune the RSNM to generate small-world and scale-free topologies, we first need to use some well-known network models to generate target network topologies. In this study, we use the Watts-Strogatz model in [25] to generate small-world network topologies and the Barabási and Albert model in [17] to generate scale-free topologies. Since these models are random models, every run of such models will generate a different topology but the associated network properties are similar between different runs. Therefore, for a given model parameter value, we need to run the models many times, each time we will get a different topology, and then we calculate the average network properties of all different topologies generated. We use such average network properties as targets to tune the RSNM. Before we run the GA, suppose that we have run the one of the models and then got the following average network properties as targets:  $T_{APL}$ ,  $T_{CC}$ , and  $T_{DD}$ , that is, target APL, target CC, and target degree distribution. Please note that  $T_{APL}$  and  $T_{CC}$  are scalars, while  $T_{DD}$  is a vector which has  $N_N$  entries, and entry  $T_{DD}(i)$  is the percentage of nodes whose degree is  $(i-1)$ ,

FIGURE 5:  $P_{DD}$ ,  $T_{DD}$ , and tolerable zone.

$i = 1, \dots, N_N$ . With  $T_{APL}$ ,  $T_{CC}$ , and  $T_{DD}$ , we can calculate the fitness of a chromosome as follows in this study:

$$f = \frac{w_1}{1 + |P_{APL} - T_{APL}|} + \frac{w_2}{1 + |P_{CC} - T_{CC}|} + \frac{w_3}{1 + E_{DD}}, \quad (16)$$

where  $w_i > 0$ ,  $i = 1, 2, 3$ , are weights which determine the contribution of each network property to the fitness,

$$E_{DD} = \sum_{i=1}^{N_N} (\max(z(i), |P_{DD}(i) - T_{DD}(i)|) - z(i))^2, \quad (17)$$

$P_{APL}$ ,  $P_{CC}$ , and  $P_{DD}$  are the network properties of the topology associated with the chromosome, and  $z(i)$  defines a tolerable zone for the gap between  $P_{DD}$  and  $T_{DD}$ .  $E_{DD}$  indicates how similar the degree distribution associated with the chromosome is to the target one (see Figure 5 for illustration). From the definition of fitness function in (16), one can see that if the topology associated with the chromosome has similar network properties to those of target topology, then the chromosome will have a large fitness.

It should be pointed out that, if we assume that all nodes have the same  $\alpha(i)$ ,  $\beta_R(i)$ , and  $\beta_L(i)$ , and all SGR have the same  $E_{SGR}(i)$ , then we have  $\alpha_L = \alpha_U$ ,  $\beta_{R,L} = \beta_{R,U}$ ,  $\beta_{L,L} = \beta_{R,U}$ , and  $E_{SGR,L} = E_{SGR,U}$ , and a chromosome will output a unique topology, whose network properties can be used straightforwardly to calculate the fitness according to (16) and (17). However, in most cases when we tune the RSNM, we allow certain RSRP to have a normal distribution with a range, for example, allowing  $\alpha(i)$  to vary within  $[\alpha_L, \alpha_U]$ , where  $\alpha_L \neq \alpha_U$ . In such a case, although a chromosome gives values for  $\alpha_L$  and  $\alpha_U$ , the output of RSNM is not unique. Therefore, to calculate the fitness of a chromosome, we need to run the RSNM for many times and then use the average network properties of all generated topologies.

## 5. Simulation Results

**5.1. Simulation Setup.** In this section, we aim to investigate, when and how the RSNM may generate small-world and scale-free network topologies. To this end, firstly, we need some models which can well generate such two categories of topologies, and as indicated in Section 4.2, we used established models from the literature [17, 25]. From the modeling

process in Section 3.2, one can easily see that, even if all RSRPs are the same, the spatial distribution of nodes can largely affect the final output topology. Therefore, in this simulation, we consider 3 different types of node distribution: circle distribution, grid distribution, and random distribution. In circle distribution, all nodes are evenly distributed along a circle whose radius is 1000. In grid distribution, all nodes are evenly and regularly arranged in a rectangular area defined by two points:  $(-1000, -1000)$  and  $(1000, 1000)$ . A random distribution is generated by introducing random disturbances to the location of nodes in a grid distribution.

The Watts-Strogatz model [25] can generate lattice topology, small-world network, and random graph by increasing the probability for randomly rewiring lattice edges, denoted as  $P_{RRLE}$  hereafter, from 0 to 1. In this simulation, we choose  $P_{RRLE} = 0, 0.08$  and 1, respectively. Then, combining a different node distribution and  $P_{RRLE}$  values, we have nine test scenarios for this model denoted from WS1 to WS9 as shown in Table 1.

In the Barabási and Albert model [17], the probability for preferential attachment (denoted by  $P_{PA}$  hereafter) plays a crucial role in generating scale-free topology. In this simulation, for node  $i$ , the  $P_{PA}$  is formulated as

$$P_{PA}(i) = \frac{D_N(i)^\tau}{\sum_{j=1}^{N_N} D_N(j)^\tau}. \quad (18)$$

Basically, the coefficient  $\tau$  determines the power law of the resulted scale-free network. Obviously, if  $\tau = 0$ , then one has an almost random graph, whilst if  $\tau$  is too large, say,  $\tau \geq 2$ , then a winner-takes-all network topology is likely to emerge. Here, we will use nine Barabási and Albert test scenarios, denoted as from BA1 to BA9, with different  $\tau$  and node distribution, as shown in Table 2.

Different sets of values for RSRPs in the RSNM might lead to similar topologies. Therefore, in the simulation, by referring to the key network properties of the associated target topology, we use the reported GA to generate six different sets of RSRP values in order to compare with the WS model, and 10 different sets of RSRP values for comparing with the BA model. Each set of RSRP values determines a specific RSNM scenario, and these 16 RSNM test scenarios are denoted as from RSNM1 to RSNM16, as listed in Tables 1 and 2. To tune the RSRP values in each RSNM test scenario, the reported GA has a population of 100, a maximal generation of 200, a mutation probability of 0.1, and a crossover probability of 0.4. For the fitness function defined by (16) and (17), we set  $w_1 = 2$ ,  $w_2 = 1$ ,  $w_3 = 5$ , and  $z(i) = 0.02$ . For the set of RSRP values given by a chromosome, the RSNM is run to generate 20 network topologies, and then the average values of the associated key network properties are used to calculate the fitness of that chromosome. The populations of chromosomes compete and evolve through generations according to their fitness. Then, the fittest chromosome in the last generation of a GA run determines the set of RSRP values for the associated RSNM test scenario. The RSRP values tuned by the reported GA for each of the 16 RSNM test scenarios are given in Tables 1 and 2. In some test scenarios, some RSRPs are assumed to be the same for all nodes, that is, have the same lower bound

TABLE 1: Comparative results: to generate small-world network topology.

NDT		Setup of model parameters	APL	ADEE	CC	ASSO
WS1	Circle dist.	$P_{RRLE} = 0$	12.8788	94.2	0.5000	0.0000
WS2	Circle dist.	$P_{RRLE} = 0.08$	4.4299	305.7	0.3117	0.0082
WS3	Circle dist.	$P_{RRLE} = 1$	3.5061	1325.0	0.0465	0.1795
WS4	Grid dist.	$P_{RRLE} = 0$	6.6667	222.2	0.0000	0.4010
WS5	Grid dist.	$P_{RRLE} = 0.08$	4.2663	365.9	0.2399	0.0636
WS6	Grid dist.	$P_{RRLE} = 1$	3.5262	1024.0	0.0549	0.2630
WS7	Random	$P_{RRLE} = 0$	6.2159	239.9	0.3487	0.0557
WS8	Random	$P_{RRLE} = 0.08$	4.3079	362.3	0.2468	0.1269
WS9	Random	$P_{RRLE} = 1$	3.5132	1079.9	0.0758	0.2356
RSNM1	Circle dist.	$\beta_R = 5.1, \beta_L = 3.1, \alpha = 440.5, N_{SGR} = 10, \text{ and } E_{SGR}(i) \in [2489.6, 3507.7]$	12.8788	94.2	0.5000	0.0000
RSNM2	Circle dist.	$\beta_R(i) \in [0.6, 1.2], \beta_L(i) \in [0.1, 0.8], \alpha(i) \in [482.4, 541.7], N_{SGR} = 10, \text{ and } E_{SGR}(i) \in [313.5, 650.0]$	8.1432	151.9	0.4836	0.0598
RSBM3	Grid dist.	$\beta_R = 4.5, \beta_L = 4.1, \alpha = 1413.7, N_{SGR} = 10, \text{ and } E_{SGR}(i) \in [3927.3, 17181.8]$	6.6667	222.2	0.0000	0.4010
RSNM4	Grid dist.	$\beta_R(i) \in [4.5, 6.2], \beta_L(i) \in [2.3, 7.5], \alpha(i) \in [1507.3, 1619.1], N_{SGR} = 10, \text{ and } E_{SGR}(i) \in [3901.2, 17215.5]$	4.9061	288.2	0.3396	0.0962
RSBM5	Random	$\beta_R = 4.5, \beta_L = 4.5, \alpha = 1319.5, N_{SGR} = 10, \text{ and } E_{SGR}(i) \in [3915.3, 17198.1]$	5.4315	240.2	0.4348	0.1031
RSNM6	Random	$\beta_R(i) \in [4.5, 6.1], \beta_L(i) \in [1.9, 7.0], \alpha(i) \in [1853.8, 1527.2], N_{SGR} = 10, \text{ and } E_{SGR}(i) \in [3897.9, 17301.2]$	4.0364	314.2	0.5542	0.1741

TABLE 2: Comparative results: to generate scale-free network topology.

NDT		Setup of model parameters	APL	ADEE	CC	ASSO
BA1	Circle dist.	$\tau = 0$	2.6561	1235.6	0.0600	0.0324
BA2	Circle dist.	$\tau = 1$	2.5103	1272.5	0.1339	0.1435
BA3	Circle dist.	$\tau = 2$	1.9350	1279.5	0.6875	0.4213
BA4	Grid dist.	$\tau = 0$	2.4652	1146.6	0.0725	0.0260
BA5	Grid dist.	$\tau = 1$	2.3690	1154.5	0.1434	0.1232
BA6	Grid dist.	$\tau = 2$	1.9224	1143.8	0.6492	0.3930
BA7	Random	$\tau = 0$	2.2271	1146.7	0.0953	0.0369
BA8	Random	$\tau = 1$	2.1768	1153.1	0.1549	0.1129
BA9	Random	$\tau = 2$	1.8991	1151.1	0.6214	0.3033
RSNM7	Circle dist.	$\beta_R = 5, \beta_L = 5.1, \alpha = 1036.7, N_{SGR} = 100, \text{ and } E_{SGR}(i) \in [0, 56549]$	2.3208	634.5	0.7718	0.3593
RSNM8	Grid dist.	$\beta_R = 5, \beta_L = 5.0, \alpha = 1036.7, N_{SGR} = 100, \text{ and } E_{SGR}(i) \in [0, 62832]$	2.1030	599.9	0.7392	0.3272
RSBM9	Random	$\beta_R = 5, \beta_L = 5.1, \alpha = 1036.7, N_{SGR} = 100, \text{ and } E_{SGR}(i) \in [0, 62832]$	2.1857	564.9	0.7397	0.3234
RSNM10	Circle dist.	$\beta_R = 100, \beta_L(i) \in [1.1, 15.2], \alpha = 1036.7, N_{SGR} = 100, \text{ and } E_{SGR} = 62832$	9.8945	213.1	0.5579	0.0916
RSBM11	Grid dist.	$\beta_R = 100, \beta_L(i) \in [1.0, 10.3], \alpha = 1036.7, N_{SGR} = 100, \text{ and } E_{SGR} = 10053$	2.0285	693.1	0.8448	0.3445
RSNM12	Random	$\beta_R = 100, \beta_L(i) \in [0.5, 10.2], \alpha = 1036.7, N_{SGR} = 100, \text{ and } E_{SGR} = 13100$	2.4119	591.7	0.7648	0.3947
RSNM13	Circle dist.	$\beta_R = 5, \beta_L = 5.1, \alpha(i) \in [439.8, 3141.6], N_{SGR} = 1, \text{ and } E_{SGR} = 2199.1$	5.0135	365.7	0.6619	0.0138
RSNM14	Grid dist.	$\beta_R = 5, \beta_L = 9.2, \alpha(i) \in [2585.0, 6911.5], N_{SGR} = 1, \text{ and } E_{SGR} = 9424.8$	2.8947	473.5	0.6374	0.1715
RSBM15	Random	$\beta_R = 5, \beta_L = 8.7, \alpha(i) \in [2585.0, 6911.5], N_{SGR} = 1, \text{ and } E_{SGR} = 9419.3$	2.1877	547.8	0.7099	0.1956
RSNM16	Random	$\beta_R = 5, \beta_L = 0.5, \alpha = 471.2, N_{SGR} = 1, \text{ and } E_{SGR} = 7854.0$	1.9594	902.1	0.9648	0.9524

as the upper bound, while other RSRPs are assumed to have a normal distribution within certain ranges. For example, in RSNM1,  $\beta_R$ ,  $\beta_L$ , and  $\alpha$  are the same for all nodes, while  $E_{SGR}(i)$  may vary within a range. Then, in RSNM, we use GA to tune  $\beta_R$ ,  $\beta_L$ ,  $\alpha$ , and the boundary values for  $E_{SGR}(i)$ .

For each test scenario, the associated network model runs for 100 times, and the average values of five important network properties, that is, the average path length (APL)

between connected nodes, the average distance of established edges (ADEE), the clustering coefficient (CC), the assortativity (ASSO), and the degree distribution (DD), are calculated and analyzed. Please note the difference between APL and ADEE: APL is measured as the number of intermediate nodes, while ADEE is defined as physical distance.

All models are coded and all tests are conducted in a MATLAB environment on a personal computer with

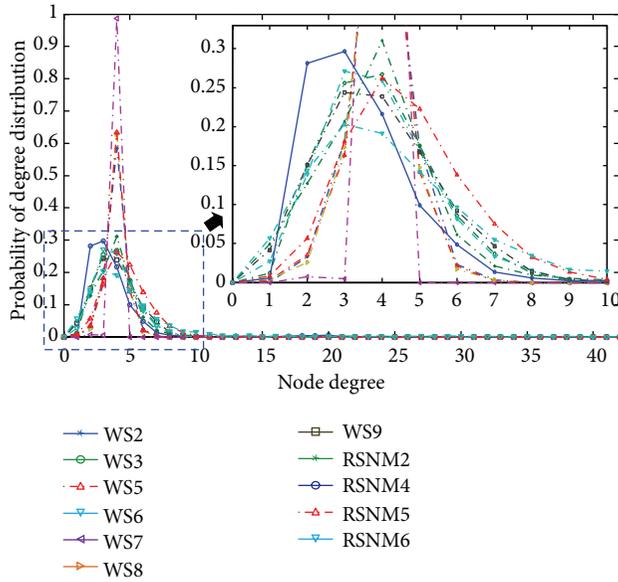


FIGURE 6: Average degree distribution of small-world networks generated by WS and RSNM.

a 2.6 GHz CPU, 4 GB memory, and the Windows 7 operating system.

**5.2. Simulation Results for the Small-World Topology.** In this subsection, we have  $N_N = 100$  nodes in each test of the simulation and then use a certain network model to establish  $N_L = 200$  edges between nodes to get a network topology. The number of SGR is always set as  $N_{SGR} = 10$  for the RSNM. The simulation results of WS1 to WS9 and RSNM1 to RSNM6 are given in Table 1 and Figure 6, and some topology examples generated during the simulation are given in Figure 7. From the table and figures, one may have the following observations.

- (i) Firstly, the data of WS1 to WS9 in Table 1 show that, compared with lattices (when  $P_{RRLE} = 0$ ) and random networks (when  $P_{RRLE} = 1$ ), a distinctive feature of small-world networks, as analyzed in [12], is that they have relatively smaller APL (compared with the associated lattices) and larger CC (compared with the associated random networks). Therefore, the first aim of this simulation is to test whether the reported GA can tune RSRPs properly to make the RSNM to generate networks with the same feature.
- (ii) From the data of RSNM2, RSNM4, RSNM5, and RSNM6, one can see that the RSNM is indeed capable of generating topologies with the above small-world network feature.
- (iii) Compared with the small-world networks generated by the WS, Table 1 also shows that the small-world networks generated by the RSNM usually have larger APL and CC (e.g., compare WS2 with RSNM2, WS5 with RSNM4, and WS8 with RSNM5 and RSNM6). This is understandable, because the

process of rewiring lattice edges in the WS is purely random, regardless of spatial distance between nodes. Due to such a random process, the WS has more edges established between spatially far away nodes, which help to reduce APL more significantly than the RSNM. Random rewiring lattice edges also lead to smaller CC when compared with the RSNM because the ripple-spreading process in the RSNM is more likely to establish edges within neighbourhood, which usually causes large CC.

- (iv) Also because of the difference between the random rewiring process in the WS and the ripple-spreading process in the RSNM, the small-world networks of RSNM usually have smaller ADEE. An interesting finding is that, when multiplying APL with ADEE, the small-world networks generated by the RSNM generally have the smaller travelling distance between two randomly chosen nodes than those of the WS actually than all networks generated by the WS including lattices and random networks.
- (v) When comparing the data of WS1 and WS4, that is, two lattice cases, with those of RSNM1 and RSNM3, one can see that the RSNM can also generate exactly the same lattices if the RSRPs are set properly. The WS initializes lattice with the spatial information of neighborhood; this explains when the RSNM can also generate lattices (the RSNM is particularly good at using spatial information to generate networks).
- (vi) Figure 6 compares the degree distribution of different models, except WS1, WS4, RSNM1, and RSNM3 (they are lattices and their degree distribution is simply a singleton). Basically, in the WS, the smaller the value of  $P_{RRLE}$ , the sharper the degree distribution, in other words, the less diversified the node degree. Overall, no matter whether we consider lattice, small-world topology, or random network, the degree distribution is similar to a Poisson distribution. In general, the degree distribution of small-world networks generated by the RSNM is more diversified than that of WS small-world networks. This is probably because the ripple-spreading process is more complicated than the random rewiring process, and therefore the node degree is less precisely managed in the RSNM than in the WS. Actually, the degree distribution of RSNM small-world networks is more like that of WS random networks (WS3, WS6, and WS9).
- (vii) Figure 7 gives some examples of small world networks generated by the WS and the RSNM (for the sake of comparison, Figure 7(d) is a random network, and Figure 7(h) is a network where edges are only established between neighborhood nodes). From Figure 7, one can see clearly that long-distance jump edges in the WS are completely random, whilst in the RSNM, they are mainly established around a few hub-like nodes. This feature, along with the degree distribution of RSNM4 in Figure 6, which is the least Poisson-distribution-like, may imply that the RSNM is more

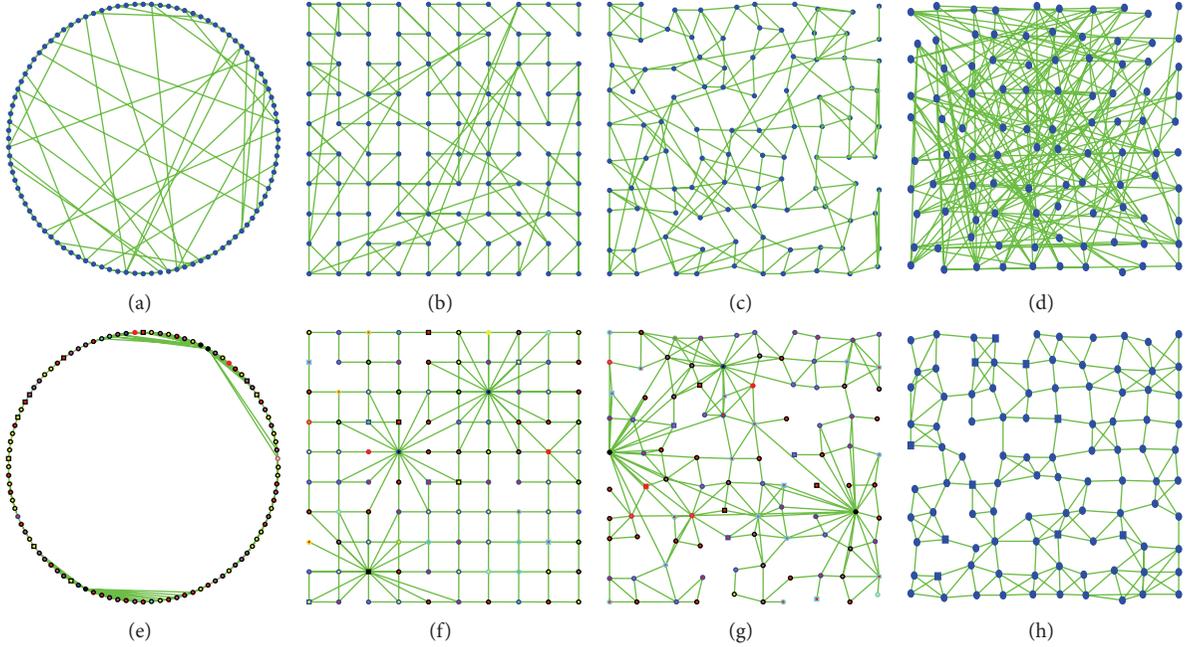


FIGURE 7: Examples of small-world networks generated by WS and RSNM: (a) WS2; (b) WS5; (c) WS8; (d) WS9; (e) RSNM2; (f) RSNM4; (g) RSNM6; (h) RSNM5.

likely a natural scale-free network model. This will be further investigated in the next section.

- (viii) In real-world networks, a node may not connect to its neighborhood nodes but some far away nodes. Figures 7(e)–7(g) clearly show that this is achievable in the RSNM, for instance, when closer nodes have large  $\beta_L$ , while some far away nodes have small enough  $\beta_L$ . However, the structure of connecting far away nodes rather than closer ones does not emerge as often in the RSNM as in the WS.
- (ix) The random network Figure 7(d) is generated by the WS. In this study, we have realized that it is very difficult to tune the RSRPs in order to generate purely random networks. Basically, because spatial distance plays a crucial role in the RSNM, we assume that it is impossible for the RSNM to generate purely random networks like Figure 7(d).

**5.3. Simulation Results about Scale-Free Topology.** In this subsection, we again have  $N_N = 100$  nodes in each test of the simulation, the number of established edges  $N_L$  in the RSNM may vary depending on how fast ripple energy decays, and then the comparative BA model is required to generate a network with the same edge number. The number of SGR is always set as  $N_{SGR} = 100$  for RSNM7 to RSNM12 and  $N_{SGR} = 1$  for RSNM13 to RSNM16. As discussed in Section 3.2, it is the ratio relationships between the RSRPs that largely determine the output topology; therefore, in this subsection, we preset  $\beta_R = 5$  for RSNM7 to RSNM9 and RSNM13 to RSNM16 and preset  $\beta_R = 100$  for RSNM10 to RSNM12. The simulation results of BA1 to BA9 and RSNM7 to RSNM16 are given in

Table 2 and Figures 8 and 9, from which one may make the following observations.

- (i) Firstly, let us analyze the networks generated by the classical scale-free model. From the data of BA1 to BA9 in Table 2, one can see clearly that, as  $\tau$  goes up from 0 to 2, the APL decreases whilst both the CC and the ASSO increases. Compared with lattices, small-world networks and random networks, such as WS1 to WS9 in Section 5.2, and scale-free networks in general have much smaller APL and larger CC and ASSO. Hub nodes in scale-free networks contribute a lot to smaller APL. In a winner-takes-all network, for example, when  $\tau \geq 2$ , a major hub node may connect to all other nodes, which means that the APL is less than 2. The large CC results from the fact that, in a scale-free network, two connected nonhub nodes are likely to connect to the same hub node, and therefore a three-node cluster is formed. In a winner-takes-all network, any two non-hub nodes connect to the hub, so the associated CC is very large. Since the ASSO is an index to indicate the preference for a network node to attach to others that are similar or different in any way [26], it of course goes up with the number of preferential attachments, which is determined by  $\tau$ .
- (ii) With properly tuned RSRPs by the reported GA, Table 2 shows that the RSNM can generate scale-free networks indeed, as the RSNM networks share the same features as the BA scale-free networks, that is, small APL and large CC and ASSO. It should be noted that, in the BA experiments, the only parameter  $\tau$  is increased gradually in order to reveal the change

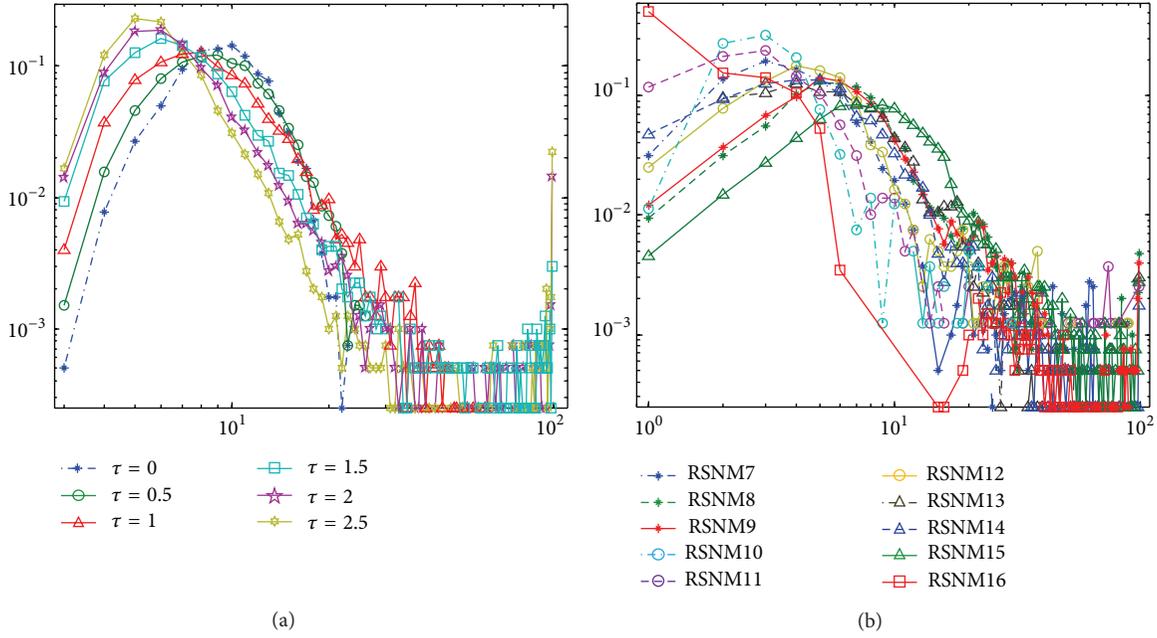


FIGURE 8: Average degree distribution of scale-free networks generated by BA and RSNM: (a) BA with random node distribution and different values for  $\tau$ ; (b) RSNMs.

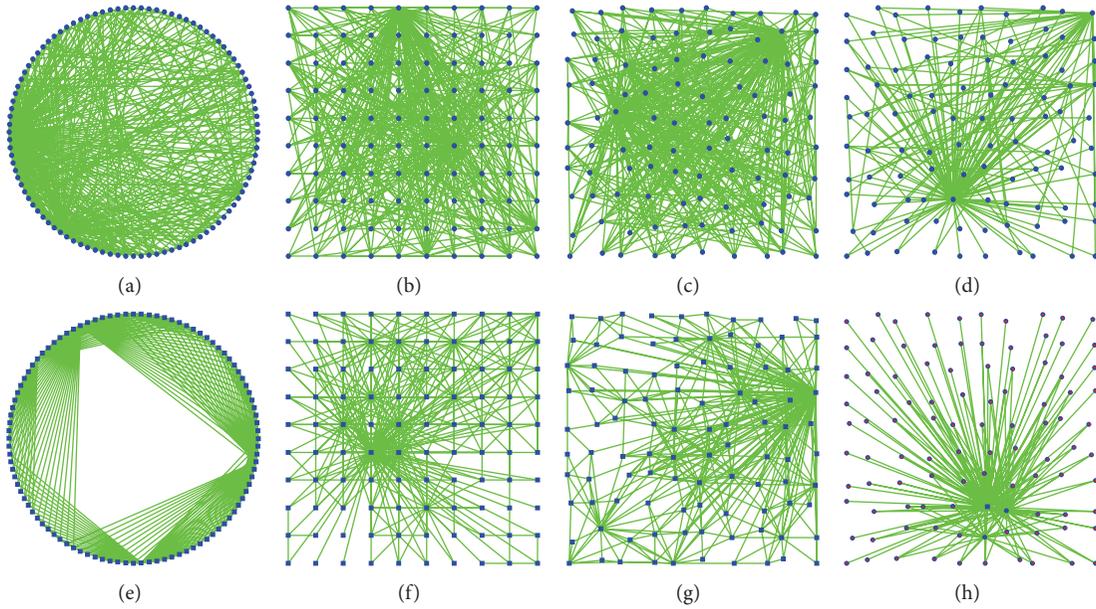


FIGURE 9: Examples of scale-free networks generated by BA and RSNM: (a) BA2; (b) BA5; (c) BA8; (d) BA9; (e) RSNM7; (f) RSNM11; (g) RSNM15; (h) RSNM16.

trend in network properties, while in the RSNM experiments, the changes in RSRPs are not continuous or even associated, and they are just tuned with purpose by referring to the BA scale-free networks.

(iii) Table 2 also shows that, in general, the RSNM networks have much smaller ASSO than the associated BA networks. This is reasonable because the BA model does not consider spatial distance at all when

establishing an edge. By multiplying the APL and ASSO, one can see again that the RSNM networks are more efficient in terms of overall travelling distance between two randomly chosen nodes. As is well known, under restricted resources, scale-free networks are highly appreciated because they are efficient in terms of overall travelling distance. For instance, given limited aircraft and staff, most airline companies now operate on hub-and-spoke airline route

networks, which is an engineering version of scale-free network [27]. Compared with the BA, the RSNM can make more of this advantage of scale-free works.

- (iv) Figure 8(a) shows how the degree distribution changes when  $\tau$  increases from 0 to 2.5 in the BA. Basically, for a scale-free network, the main part of the right side slope of the degree distribution appears linear in a log-log plot, and the slope becomes sharper as  $\tau$  increases. Figure 8(b) gives the degree distribution of RSNM7 to RSNM16, from which one can clearly see they have approximately linear right side slope with a different sharpness. This further confirms that the RSNM can generate scale-free networks, given that the RSRPs are properly tuned.
- (v) Figure 9 gives some examples of scale-free networks generated by the BA and the RSNM. Apparently, the BA scale-free networks are more spatially chaotic, whilst the RSNM networks are much better organized in terms of spatial view.
- (vi) One may notice that, in Table 2, the APL of RSNM10 and RSNM13 is much larger than others. This is mainly because of the circle node distribution used. In a scale-free network, it often needs to pass a hub node to travel between two nodes. In the RSNM, even a hub node tends to establish edges locally. Therefore, in the case of circle node distribution, sometimes it may need to pass through quite a few hub nodes around the circle to travel between two nodes.

## 6. Conclusions and Future Work

This paper is concerned with how to apply a genetic algorithm (GA) to tune the parameters of ripple-spreading network model (RSNM), in order to generate small-world and scale-free network topologies. As a newly reported complex network model, the RSNM may well describe the development of many real-world network systems where the ripple-spreading effect of some local events' influence plays an important role. However, a demerit is that those ripple-spreading related parameters (RSRPs) of RSNM have no obvious or simple relationship with the network properties of output topology. In particular, since small-world and scale-free topologies are two of the most important features of many real-world complex network systems, it is crucial to understand, when and how the RSNM can generate these two categories of network topologies. This paper develops an effective GA to tune the RSRPs in the RSNM, and the simulation shows that the RSNM has a great flexibility in generating various network topologies. The reported work further verifies the theoretical potential of RSNM. Therefore, future work should pay more attention to real applications of the RSNM (such as in the study of disaster management and epidemic dynamics), and the GA developed in this paper can then be easily extended to tune model parameters, which will play an important role in simulating a real-world system. For example, when applying the RSNM to study a specific real-world network system, comparison should be conducted between the reported GA

and existing relevant methods for that specific system in terms of time performance and scalability.

## Acknowledgments

This work was supported in part by the China "973" Project under Grant 2012CB955404, the Project Grant 2012-RC-02 from Beijing Normal University, China, and the Seventh Framework Programme (FP7) of the European Union under Grant PIOF-GA-2011-299725.

## References

- [1] J. H. Holland, *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor, Mich, USA, 1975.
- [2] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, Prentice Hall, New York, NY, USA, 3rd edition, 2010.
- [3] T. Back, U. Hammel, and H.-P. Schwefel, "Evolutionary computation: comments on the history and current state," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 3–17, 1997.
- [4] X.-B. Hu, M. Wang, M. S. Leeson, E. L. Hines, and E. Di Paolo, "Deterministic ripple-spreading model for complex networks," *Physical Review E*, vol. 83, no. 4, Article ID 046123, 14 pages, 2011.
- [5] P. Erdős and A. Rényi, "On random graphs. I," *Publicationes Mathematicae Debrecen*, vol. 6, pp. 290–297, 1959.
- [6] F. Liljeros, C. R. Edling, L. A. Nunes Amaral, H. E. Stanley, and Y. Åberg, "Social networks: the web of human sexual contacts," *Nature*, vol. 411, no. 6840, pp. 907–908, 2001.
- [7] I. Frommer and G. Pundoor, "Small-worlds: a review of recent books," *Networks*, vol. 41, no. 3, pp. 174–180, 2003.
- [8] W. Aiello, F. Chung, and L. Lu, "A random graph model for massive graphs," in *Proceedings of the 32nd Annual ACM Symposium on Theory of Computing*, pp. 171–180, ACM, New York, NY, USA, 2000.
- [9] M. E. J. Newman, "Clustering and preferential attachment in growing networks," *Physical Review E*, vol. 64, no. 2, Article ID 025102(R), 4 pages, 2001.
- [10] L. Barnett, E. Di Paolo, and S. Bullock, "Spatially embedded random networks," *Physical Review E*, vol. 76, no. 5, Article ID 056115, 18 pages, 2007.
- [11] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D.-U. Hwang, "Complex networks: structure and dynamics," *Physics Reports*, vol. 424, no. 4-5, pp. 175–308, 2006.
- [12] R. Albert and A.-L. Barabási, "Statistical mechanics of complex networks," *Reviews of Modern Physics*, vol. 74, no. 1, pp. 47–97, 2002.
- [13] M. Kochen, Ed., *The Small World*, Ablex, Norwood, NJ, USA, 1989.
- [14] R. Albert, H. Jeong, and A.-L. Barabási, "Diameter of the world-wide web," *Nature*, vol. 401, no. 6749, pp. 130–131, 1999.
- [15] M. Faloutsos, P. Faloutsos, and C. Faloutsos, "On power-law relationships of the internet topology," *Computer Communication Review*, vol. 29, p. 251, 1999.
- [16] H. Jeong, B. Tombor, R. Albert, Z. N. Oltval, and A.-L. Barabási, "The large-scale organization of metabolic networks," *Nature*, vol. 407, no. 6804, pp. 651–654, 2000.
- [17] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, 1999.

- [18] M. Minoux, "Network synthesis and optimum network design problems: models, solution methods and applications," *Networks*, vol. 19, no. 3, pp. 313–360, 1989.
- [19] A. E. Eiben and M. Schoenauer, "Evolutionary computing," *Information Processing Letters*, vol. 82, no. 1, pp. 1–6, 2002.
- [20] Z.-H. Xin and H.-J. Zhang, "Neural network and genetic algorithms for topology optimization of the CCS7 network," *International Transactions in Operational Research*, vol. 9, no. 4, pp. 427–436, 2002.
- [21] E.-S. M. El-Alfy, "MPLS network topology design using genetic algorithms," in *Proceedings of IEEE International Conference on Computer Systems and Applications*, pp. 1059–1065, March 2006.
- [22] H. Liu, X.-B. Hu, S. Yang, K. Zhang, and E. Di Paolo, "Application of complex network theory and genetic algorithm in airline route networks," *Transportation Research Record*, vol. 2214, pp. 50–58, 2011.
- [23] X.-B. Hu and E. A. di Paolo, "A ripple-spreading genetic algorithm for the aircraft sequencing problem," *Evolutionary Computation*, vol. 19, no. 1, pp. 77–106, 2011.
- [24] J. Page, P. Poli, and W. B. Langdon, "Smooth uniform crossover with smooth point mutation in genetic programming: a preliminary study," in *Proceedings of the 2nd European Workshop on Genetic Programming (EuroGP '99)*, Göteborg, Sweden, May 1999.
- [25] D. J. Watts and S. H. Strogatz, "Collective dynamics of "small-world" networks," *Nature*, vol. 393, no. 6684, pp. 440–442, 1998.
- [26] M. E. J. Newman, "Assortative mixing in networks," *Physical Review Letters*, vol. 89, no. 20, Article ID 208701, 4 pages, 2002.
- [27] G. Burghouwt, J. Hakfoort, and J. R. van Eck, "The spatial configuration of airline networks in Europe," *Journal of Air Transport Management*, vol. 9, no. 5, pp. 309–323, 2003.

## Research Article

# Tundish Cover Flux Thickness Measurement Method and Instrumentation Based on Computer Vision in Continuous Casting Tundish

Meng Lu, He Qing, Xie Zhi, Yang Weimin, Ci Ying, Zhang Chuanyi, and Gao Hongliang

College of Information Science and Engineering, Northeastern University, Shenyang 110000, China

Correspondence should be addressed to Meng Lu; [menglu1982@gmail.com](mailto:menglu1982@gmail.com)

Received 10 April 2013; Accepted 2 July 2013

Academic Editor: Praveen Agarwal

Copyright © 2013 Meng Lu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Thickness of tundish cover flux (TCF) plays an important role in continuous casting (CC) steelmaking process. Traditional measurement method of TCF thickness is single/double wire methods, which have several problems such as personal security, easily affected by operators, and poor repeatability. To solve all these problems, in this paper, we specifically designed and built an instrumentation and presented a novel method to measure the TCF thickness. The instrumentation was composed of a measurement bar, a mechanical device, a high-definition industrial camera, a Siemens S7-200 programmable logic controller (PLC), and a computer. Our measurement method was based on the computer vision algorithms, including image denoising method, monocular range measurement method, scale invariant feature transform (SIFT), and image gray gradient detection method. Using the present instrumentation and method, images in the CC tundish can be collected by camera and transferred to computer to do imaging processing. Experiments showed that our instrumentation and method worked well at scene of steel plants, can accurately measure the thickness of TCF, and overcome the disadvantages of traditional measurement methods, or even replace the traditional ones.

## 1. Introduction

As the development of high-speed casting, tundish cover flux (TCF) is becoming the key material for continuous casting process, which has more and more important and irreplaceable roles on the quality of steel billet. The main functions of TCF are as follows: isolate the steel surface to prevent freezing, protect the steel surface from oxidation, absorb inclusions that are transported to the surface, maintain a low friction between steel shell and mold by lubrication, create an optimum heat transfer from steel to mold, and so on [1–4].

The melting point of TCF (about 1000~1150°C) is lower than the temperature of molten steel (about 1500°C). And the TCF is divided into three main parts according to the physical form, namely, liquid slag layer, sintered layer, and powder slag layer (shown in Figure 1). TCF is added on the top of the molten steel in the continuous casting mold, which partially melts and forms a liquid slag layer (about 6~15 mm) above the molten steel. The liquid slag layer can partly prevent the heat

of molten steel from transferring; therefore, above the liquid slag layer, temperature drops, which forms a sintered layer (about 600~900°C). Above the sintered layer, more heat is prevented; therefore, the TCF can keep its original pulverous characteristic to form a powder slag layer. The powder slag layer covers the molten steel surface evenly to prevent heat radiation and isolate oxygen.

In the continuous casting steelmaking process, due to crystallizer vibration and solidified shell movement, the liquid slag layer may slowly get into the gap between the shell and the copper wall of crystallizer. Therefore, along with the casting process, the liquid slag layer is constantly consumed by the molten steel and gets supplement from the sintered layer. Similarly, the sintered layer is constantly transformed into liquid slag layer and gets supplement from the powder slag layer. And the operator keeps an eye on the powder slag layer and adds new TCF into CC tundish as necessary. The thickness of TCF has a great effect on the continuous casting steelmaking process; if the TCF is too thin, the steel

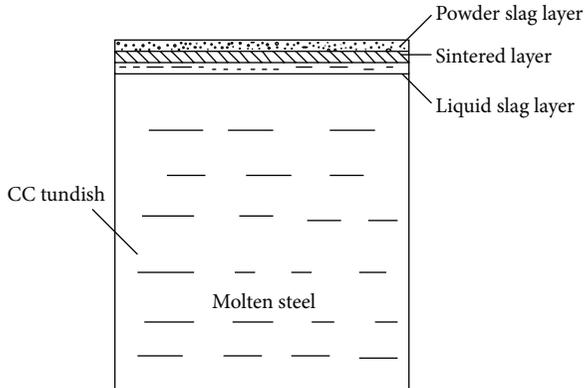


FIGURE 1: Schematic diagram of TCF three main components in the CC tundish, which are powder slag layer, sintered layer, and liquid slag layer.

will become carburized, unmelted slag patches will occur as surface defects, and worse of all the result can be a breakout because of too high friction between steel shell and mould. On the contrary, if the TCF is too thick, crystallized rim will emerge at the meniscus which will disturb the solidification process and cause surface defects on the steel. Therefore, TCF thickness is an essential parameter to guarantee the quality of strand surface and can provide important information for the operators to keep track of the status in the CC tundish.

Nowadays, in continuous casting steelmaking process, the most common used measurement methods for TCF thickness are single-wire measurement method and double-wire measurement method, because these two methods are cheap and easily operated. In single-wire measurement method, one iron wire is vertically inserted into the molten steel, and kept steady for 3~5 seconds. Then pull it out and measure the TCF thickness based on the length of color change and slag-adhering part (shown in Figure 2). In double-wire measurement method, one iron wire and one copper wire, which have the same length, are parallel and firmly tied together, then vertically inserted into the molten steel, and kept steady for 3~5 seconds. Melting point of the iron wire is higher than the temperature of liquid slag layer and lower than the temperature of the molten steel, therefore, the iron wire below the liquid slag layer is melted. Melting point of the copper wire is higher than the temperature of sintered layer and lower than the temperature of liquid slag layer; therefore, the copper wire below the sintered layer is melted. The whole thickness of TCF can be calculated based on the length of the color change and slag-adhering part on the iron wire. Compared with the single-wire measurement method, the double-wire measurement method can specifically measure the thickness of the liquid slag layer using the length of the remaining iron wire minus the length of the remaining copper wire (shown in Figure 3). Although the single-wire measurement method and the double-wire measurement method are widely used in continuous casting steelmaking process, they both have insurmountable and severe shortcomings: (1) operators need to stand on the tundish to hold the wires, which means

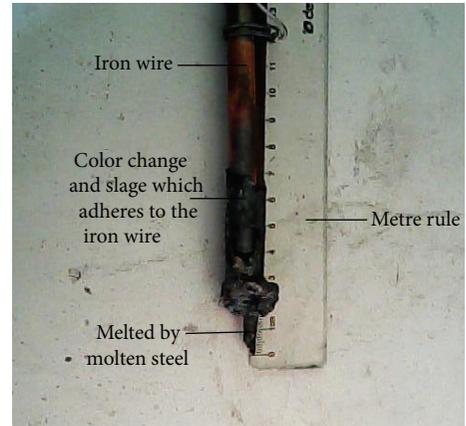


FIGURE 2: Single-wire measurement method of TCF thickness. TCF thickness can be measured by the length of color change and slag-adhering part on the iron wire.

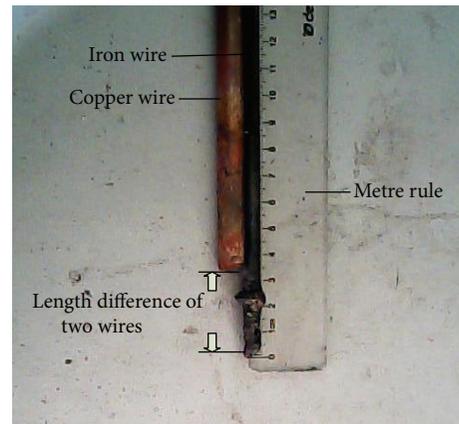


FIGURE 3: Double-wire measurement method of TCF thickness. The whole thickness of the TCF can be measured by the length of the color change and slag-adhering part on the iron wire. The thickness of the liquid slag layer can be measured by the length difference of two wires.

lethal danger to operators; (2) the detection results are easily influenced by the operators, such as hand trembling or not vertically holding the wires; (3) poor repeatability. All these shortcomings make these two TCF thickness measurement methods inaccurate and dangerous.

In this paper, a novel TCF thickness measurement method and instrumentation are proposed. We use a high-definition industrial camera to collect the image information in the CC tundish, which is fastened to a custom-designed mechanical device. The thickness of TCF is calculated based on the computer vision algorithms, including image denoise method, monocular range measurement method, scale invariant feature transform (SIFT), and image gray gradient detection method. Our measurement method is noncontact measurement, full automatic working, good repetitiveness, and absolutely safe.

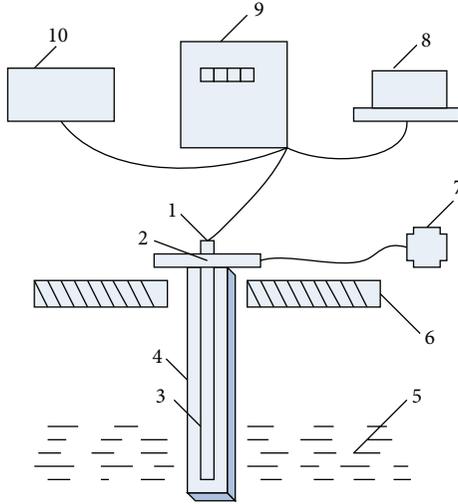


FIGURE 4: Schematic diagram of the continuous temperature measurement system based on the blackbody radiation theorem. (1) Temperature sensor, (2) optical lens, (3) internal tube, (4) external tube, (5) molten steel, (6) tundish, (7) bracket, (8) single-chip microcontroller (SCM), (9) signal processor, and (10) display for temperature.

## 2. Method

The TCF thickness measurement method and instrumentation are an extension of a previous work from our group [5–21], which is continuous temperature measurement method and sensor for molten steel in CC tundish based on blackbody radiation theorem. We briefly recall it in order to make the following sections more clearly.

Theoretical principle of continuous temperature measurement method is blackbody radiation theorem:

$$E_b = (\lambda, T_L) = \varepsilon^T(\lambda, T) \cdot E_b(T_0, \lambda). \quad (1)$$

$E_b$  represented the spectral radiosity of blackbody chamber,  $\varepsilon^T$  represented the spectral emissivity of blackbody chamber,  $T_L$  represented the brightness temperature of blackbody chamber, and  $T$  represented the actual temperature of blackbody chamber.

Based on this theorem, we specifically made a measurement bar and used it as a temperature sensor, which was inserted into the molten steel to measure the temperature of molten steel. The measurement bar was the core component of the system, which was composed of external tube and internal tube. The external tube was heat resistant, shock resistant, and anti corrosion and had good heat conduction, which can offer protection for the internal tube. The internal tube was made of certain translucent medium, which had steady radiation characteristic and specular/diffusion reflection characteristic. Heat radiation in the measurement bar was transformed into electrical signal by a photoelectric transducer. And the electrical signal can be used to calculate the temperature of the molten steel based on the blackbody radiation theorem. The schematic diagram of the system was shown in Figure 4.

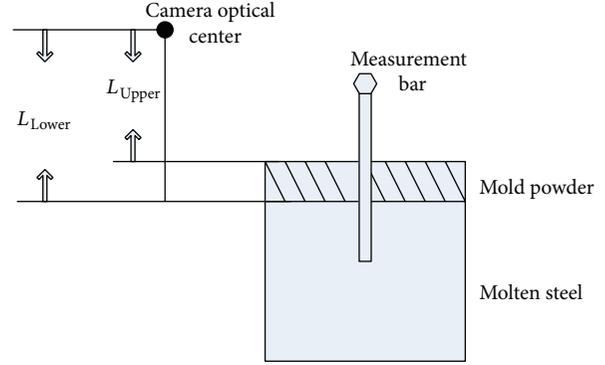


FIGURE 5: Schematic diagram of measurement bar, which is inserted into the molten steel.

More details about the continuous temperature method and sensor for molten steel in CC tundish can be referenced from [5–13].

**2.1. Theory.** The measurement bar is also the core component of TCF thickness measurement method and instrumentation. While the measurement bar is inserted into the molten steel, a high-definition industrial camera is used to collect the image information in the tundish. Based on computer vision algorithms, which are elaborated in the Sections 2.3 and 2.4, TCF thickness is calculated by

$$L_{\text{powder}} = L_{\text{lower}} - L_{\text{upper}}, \quad (2)$$

where  $L_{\text{lower}}$  represents the distance from TCF lower surface to optical center of the camera and  $L_{\text{upper}}$  represents the distance from TCF upper surface to optical center of the camera (shown in Figure 5).

Molten steel and TCF have different thermal conductivities; therefore, these two different media have different temperatures, and there is obvious temperature gradient in the interface layer of these two media. While the measurement bar is inserted into the molten steel for long enough time (shown in Figure 5), the measurement bar can be in the state of thermal balance. And then the measurement bar is pulled up; the temperature information of molten steel and TCF can be reflected by the luminance of the measurement bar (shown in Figure 6). In Figure 6(a), the brighter zone indicates that this part of the measurement bar is in the molten steel before being pulled up, and the darker zone indicates that this part of the measurement bar is in the TCF before being pulled up. The arrow indicates the peak value of image gradient between the brighter zone and the darker zone, which is also the interface layer between the molten steel and the TCF. In Figure 6(b), it can be seen that the temperature on the measurement bar has an intense change, which causes the peak value of the temperature gradient indicated by the arrow. These two arrows in both Figures 6(a) and 6(b) actually point to the same location of the measurement bar, which is the lower surface of the TCF. Therefore, the distance from TCF lower surface to optical center of the camera can be calculated by gray gradient detection in the image, which is elaborated in Section 2.4.

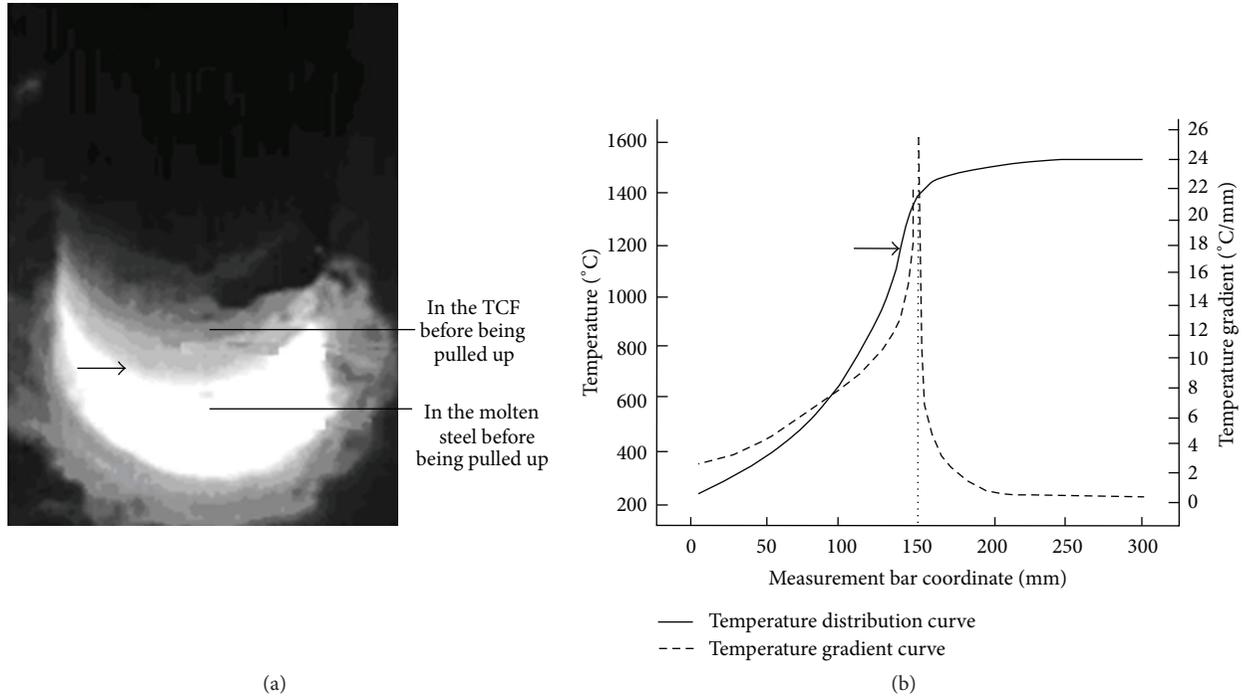


FIGURE 6: The surface condition of the measurement bar, when the measurement bar is pulled up from the molten steel. (a) Luminance information on the measurement bar, on which obvious image gray scale difference can be seen. (b) Schematic diagram of temperature distribution and gradient curve on the measurement bar. Horizontal ordinate represents coordinates of the measurement bar, left vertical ordinate represents the temperature of the measurement bar, and the right vertical ordinate represents the temperature gradient of the measurement bar.

2.2. *Instrumentation.* As mentioned before, the measurement bar is inserted into the molten steel; therefore, a specialized mechanical device is required to hold the measurement bar, and the camera is also fixed on the mechanical device (shown in Figure 7). The mechanical device can ascend and descend under the control of frequency conversion motor, and the measurement bar can be lifted and lowered with the movement of the mechanical device. The camera is fixed and well-secured on the crossbeam of the mechanical device, and in case of overheating, the camera is cooled down by cool air through ventilating pipe. With the help of this mechanical device, the height of the crossbeam, the temperature of the measurement bar, and images collected by the camera are transferred to the computer and the programmable logic controller (PLC, Siemens S7-200). We can use all these information to make judgments and calculate TCF thickness. The main functions of the computer are imaging processing and displaying the calculation result of TCF thickness. The main functions of the PLC are controlling the movement of the mechanical device and making it accurate and steady. The schematic diagram of the whole instrumentation system is shown in Figure 8.

2.3. *TCF Upper Surface Distance Measurement.* The distance from TCF upper surface to the optical center of camera is calculated based on the principle of monocular distance measurement [22, 23]. Two images  $I_1, I_2$  can be obtained while the camera is moved from position  $P_1$  to position  $P_2$ . In

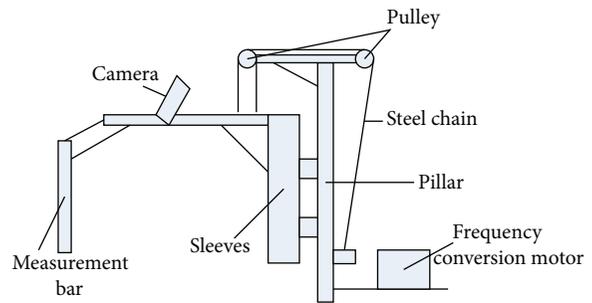


FIGURE 7: Schematic diagram of mechanical device, which holds the camera and measurement bar.

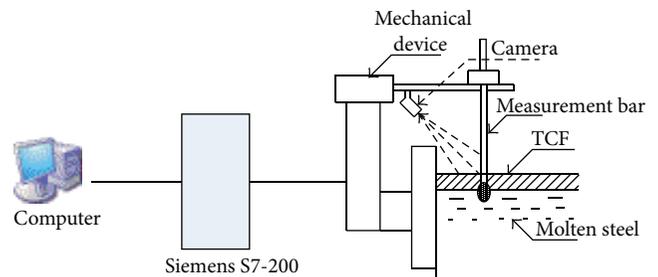


FIGURE 8: Schematic diagram of the whole instrumentation system, including computer, Siemens S7-200, mechanical device, camera, and measurement bar.

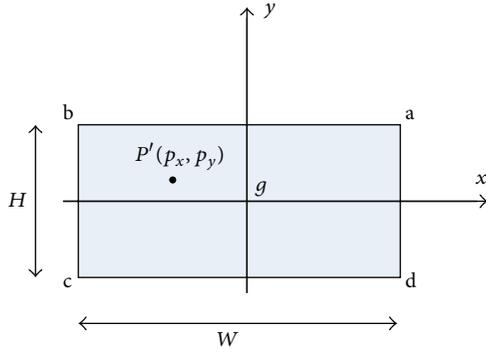


FIGURE 9: Schematic diagram of image coordinate system.  $g$  is the base point, which is in the center of the image,  $P'$  represents one pixel in the image with coordinate  $(p_x, p_y)$ ,  $x$ - $g$ - $y$  represents the rectangular coordinate system,  $W$  represents the width of the image, and  $H$  represents the height of the image.

these two images, the same objects have different coordinates due to the camera movement. According to the coordinate differences and the movement distance from  $P_1$  to  $P_2$ , the distance from the objects to the optical center of camera can be calculated. And there are two key steps in the calculation: (1) the transformation from image coordinates to the world coordinates; (2) matching the corresponding pixels of the same objects in two images  $I_1$  and  $I_2$ .

**2.3.1. Transformation from the Image Coordinates to the World Coordinates.** Suppose that the image coordinates are defined as  $(p_x, p_y)$ , and the world coordinates are defined as  $(P_x, P_y, P_z)$ . We can see that the image coordinates are two-dimensional and the world coordinates are three-dimensional; therefore, one of the world coordinates should be set as constant. The base point of the image coordinate system is defined as the center of the image (shown in Figure 9). The base point of the world coordinate system is defined as the intersection point of TCF upper surface and the optical axis originated from the camera's optical center (shown in Figure 10); therefore,  $P_z \equiv 0$ . The transformation from image coordinates  $(p_x, p_y)$  to the world coordinates  $(P_x, P_y, P_z)$  is defined as formula (3), and the details of proof can be obtained in appendix.

In Figure 10, plane  $ABU$  represents TCF upper surface,  $ABCD$  represents the camera's field of view, point  $O$  represents the camera's optical center, line  $OG$  represents the camera's optical axis, point  $G$  represents the intersection point of optical axis and plane  $ABU$ , point  $G$  is also the base point of the world coordinate system, point  $I$  represents the projection point of optical center  $O$  on the plane  $ABU$ , and the distance from point  $O$  to point  $I$  is  $h$ .  $X$ - $G$ - $Y$  represents the rectangular coordinate system.  $2\alpha_0$  represents the camera's vertical field angle,  $2\beta_0$  represents the camera's horizontal field angle, and  $\gamma_0$  represents the camera's pitch angle. Namely,  $\angle EOF = 2\alpha_0$ , line  $OG$  equally divides  $\angle EOF$ ;  $\angle KOJ = 2\beta_0$ , line  $OG$  equally divides  $\angle KOJ$ ,  $\angle GOI = \gamma_0$ . Point  $P$  is any point on the TCF upper surface, and its world coordinate is  $(P_x, P_y)$  in the rectangular coordinate system  $X$ - $G$ - $Y$ . From point  $P$ , draw

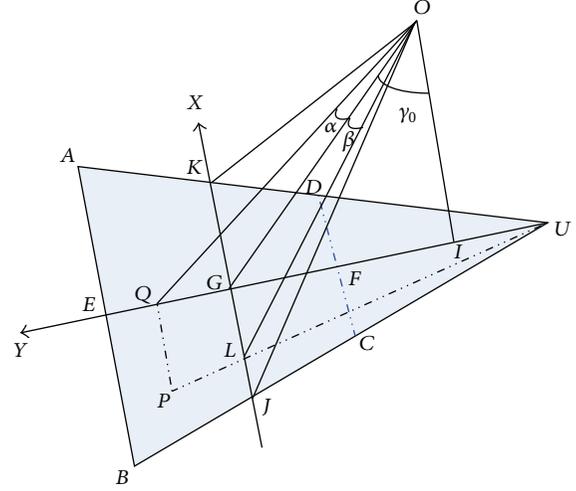


FIGURE 10: Schematic diagram of world coordinate system.

perpendicular line on the axis  $GY$ , and the perpendicular foot is point  $Q$ ; therefore,  $P_y = GQ$ . Connect point  $P$  and point  $U$ , and line  $PU$  intersects axis  $GX$  at point  $L$ ; therefore,  $P_x = GL$ . Connect point  $L$  and optical center  $O$ , and the intersection angle of line  $OL$  and optical axis is  $\beta$ . Connect point  $Q$  and optical center  $O$ , and the intersection angle of line  $OQ$  and optical axis is  $\alpha$ :

$$\alpha = \arctg \frac{2p_y \times tg\alpha_0}{H},$$

$$\beta = \arctg \frac{2p_x \times tg\beta_0}{W},$$

$$P_x = \frac{h}{\cos \gamma_0} \times tg\beta \times \left( 1 + \frac{(tg(\gamma_0 + \alpha) - tg\gamma_0) \times (\cos(\gamma_0 - \alpha_0) - \cos \gamma_0)}{(tg\gamma_0 - tg(\gamma_0 - \alpha_0)) \times \cos(\gamma_0 - \alpha_0)} \right),$$

$$P_y = h(tg(\gamma_0 + \alpha) - tg\gamma_0),$$

$$P_z = 0.$$

(3)

$H$  represents the height of the image,  $W$  represents the width of the image,  $h$  represents the vertical distance from camera's optical center to the TCF upper surface,  $2\alpha_0$  represents the camera's vertical field angle,  $2\beta_0$  represents the camera's horizontal field angle, and  $\gamma_0$  represents the camera's pitch angle.

Based on formula (3), the world coordinates of any point on the TCF upper surface can be calculated, and the only unknown parameter is  $h$ , which is the distance from the TCF upper surface to the camera's optical center. By moving the camera vertically from  $h_1$  to  $h_2$  ( $h_2 > h_1$ ), although  $h_1$  and  $h_2$  are unknown,  $\Delta h = h_1 - h_2$  can be obtained (shown in Figure 11). Suppose that the same object on the TCF upper surface has one world coordinate  $P_1(a, b)$  before movement

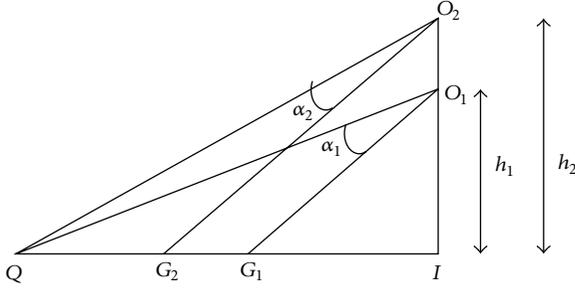


FIGURE 11: Schematic diagram of moving camera from  $h_1$  to  $h_2$ . While the camera is moved from  $h_1$  to  $h_2$ , then the camera's optical center is moved from  $O_1$  to  $O_2$  after movement; therefore, the base point of world rectangular coordinate system is also moved from  $G_1$  to  $G_2$ .

and another world coordinate  $P_2(c, d)$  after movement. While the camera is moved from  $h_1$  to  $h_2$ , then the camera's optical center is moved from  $O_1$  to  $O_2$  after movement; therefore, the base point of world rectangular coordinate system is also moved from  $G_1$  to  $G_2$ . Because the movement of camera is vertical, two world rectangular coordinate systems have the same  $y$ -axis and different  $x$ -axis; therefore,  $P_1$  and  $P_2$  have the same horizontal coordinates in the world coordinate system, namely,  $a = c$ . Therefore, the distance from TCF upper surface to camera optical center can be calculated by formula (4):

$$\begin{aligned} & \frac{h_1}{\cos \gamma_0} \times \operatorname{tg} \beta_1 \\ & \times \left( 1 + \frac{(\operatorname{tg}(\gamma_0 + \alpha_1) - \operatorname{tg} \gamma_0) \times (\cos(\gamma_0 - \alpha_0) - \cos \gamma_0)}{(\operatorname{tg} \gamma_0 - \operatorname{tg}(\gamma_0 - \alpha_0)) \times \cos(\gamma_0 - \alpha_0)} \right) \\ & = \frac{h_2}{\cos \gamma_0} \times \operatorname{tg} \beta_2 \\ & \times \left( 1 + \frac{(\operatorname{tg}(\gamma_0 + \alpha_2) - \operatorname{tg} \gamma_0) \times (\cos(\gamma_0 - \alpha_0) - \cos \gamma_0)}{(\operatorname{tg} \gamma_0 - \operatorname{tg}(\gamma_0 - \alpha_0)) \times \cos(\gamma_0 - \alpha_0)} \right), \\ & h_2 - h_1 = \Delta h, \\ & \alpha_1 = \operatorname{arctg} \frac{2p_{y1} \times \operatorname{tg} \alpha_0}{H}, \\ & \beta_1 = \operatorname{arctg} \frac{2p_{x1} \times \operatorname{tg} \beta_0}{W}, \\ & \alpha_2 = \operatorname{arctg} \frac{2p_{y2} \times \operatorname{tg} \alpha_0}{H}, \\ & \beta_2 = \operatorname{arctg} \frac{2p_{x2} \times \operatorname{tg} \beta_0}{W}. \end{aligned} \quad (4)$$

2.3.2. *Matching the Corresponding Pixels of the Same Objects in Two Images.* SIFT based image registration method is used to match the corresponding pixels of the same objects

in two images. SIFT was published by Lowe in 1999 [24] and improved in 2004 [25], which was used to detect and describe local image features. SIFT is an excellent feature descriptor, because it is invariant to uniform scaling, orientation, and partially invariant to affine distortion and illumination changes. SIFT's application includes object recognition, robotic mapping and navigation, image stitching, 3D modeling, gesture recognition, video tracking, individual identification of wildlife, and match moving.

Scale space is a formal theory for handling image structures at different scales from physical and biological vision, by representing an image as a one-parameter family of smoothed images. The main type of scale-space is the linear (Gaussian) scale-space, which can be defined by

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y). \quad (5)$$

$I(x, y)$  represents one image,  $*$  represents convolution, and  $G(x, y, \sigma)$  represents Gaussian filter function:

$$G(x, y, \sigma) = \frac{(1/2\pi\sigma^2) e^{-(x^2+y^2)}}{2\sigma^2}. \quad (6)$$

$(x, y)$  represents image coordinates, and  $\sigma$  represents scale level. Therefore, difference of Gaussian (DoG) scale space can be defined as

$$\begin{aligned} D(x, y, \sigma) &= (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \\ &= L(x, y, k\sigma) - L(x, y, \sigma). \end{aligned} \quad (7)$$

Once DoG images have been obtained, keypoints are identified as local minima/maxima of the DoG images across scales. This is done by comparing each pixel in the DoG images to its eight neighbors at the same scale and nine corresponding neighboring pixels in each of the neighboring scales. If the pixel value is the maximum or minimum among all compared pixels, it is selected as a candidate keypoint.

Scale-space extreme detection produces too many keypoint candidates, some of which are unstable. The next step in the algorithm is to perform a detailed fit to the nearby data for accurate location, scale, and ratio of principal curvatures. This information allows points to be rejected that have low contrast (and are therefore sensitive to noise) or are poorly localized along an edge.

The interpolation of keypoints is done using the quadratic Taylor expansion of the DoG scale-space function:

$$D(X) = D + \frac{\partial D^T}{\partial X} X + \frac{1}{2} X^T \frac{\partial^2 D}{\partial X^2} X. \quad (8)$$

Then, the location of the extreme  $\hat{X}$  is determined by taking the derivative of this function with respect to  $X$  and setting it to zero:

$$\begin{aligned} \hat{X} &= -\frac{\partial^2 D^{-1} \partial D}{\partial X^2 \partial X}, \\ D(\hat{X}) &= D + \frac{1}{2} \frac{\partial D^T}{\partial X} \hat{X}. \end{aligned} \quad (9)$$

The DoG function will have strong responses along edges, even if the candidate keypoint is not robust to small amounts of noise. Therefore, in order to increase stability, the Hessian matrix is used to eliminate the keypoints that have poorly determined locations but have high edge responses:

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix},$$

$$\text{Tr}(H) = D_{xx} + D_{yy} = \alpha + \beta, \quad (10)$$

$$\text{Det}(H) = D_{xx}D_{yy} - (D_{xy})^2 = \alpha\beta.$$

$\alpha$  represents bigger eigenvalue, and  $\beta$  represents smaller eigenvalue. Suppose that  $\alpha = r\beta$ . We can get

$$R = \frac{\text{Tr}(H)^2}{\text{Det}(H)} = \frac{(\alpha + \beta)^2}{\alpha\beta} = \frac{(r\beta + \beta)^2}{r\beta^2} = \frac{(r + 1)^2}{r}. \quad (11)$$

It follows that, for some threshold eigenvalue ratio  $r_{\text{th}}$ , if  $R$  for a candidate keypoint is larger than  $(r_{\text{th}} + 1)^2/r_{\text{th}}$ , that keypoint is poorly localized and hence rejected.

Each keypoint is assigned one or more orientations based on local image gradient directions. This is the key step in achieving invariance to rotation as the keypoint descriptor can be represented relative to this orientation and therefore achieve invariance to image rotation:

$$m(x, y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2}$$

$$\theta(x, y) = \tan^{-1} \left( \frac{L(x, y+1) - L(x, y-1)}{L(x+1, y) - L(x-1, y)} \right) \quad (12)$$

$m(x, y)$  represents the gradient magnitude, and  $\theta(x, y)$  represents the orientation.

A set of orientation histograms are created on  $4 \times 4$  pixel neighborhoods with 8 bins each. These histograms are computed from magnitude and orientation values of samples in a  $16 \times 16$  region around the keypoint such that each histogram contains samples from a  $4 \times 4$  subregion of the original neighborhood region. The magnitudes are further weighted by a Gaussian function with  $\sigma$  equal to one half the width of the descriptor window. The descriptor then becomes a vector of all the values of these histograms. Since there are  $4 \times 4 = 16$  histograms each with 8 bins, the vector has 128 elements. This vector is then normalized to unit length in order to enhance invariance to affine changes in illumination.

The SIFT feature descriptors extracted from two images  $I_1, I_2$  while the camera is moved from position  $P_1$  to position  $P_2$  are shown in Figure 12.

**2.4. TCF Lower Surface Distance Measurement.** While the measurement bar is pulled out, due to the temperature difference, there is temperature gradient on the measurement bar. And the temperature gradient is reflected as gray gradient in the image (shown in Figure 6). Therefore, the distance from TCF lower surface to the camera optical center is calculated based on gray gradient detection in the image.

The measurement bar and the camera are both fixed in the mechanical device, so they are in the relatively static state. Therefore, in the image, pixels in the middle line of measurement bar have constant distance to the camera optical center. Suppose that  $(L_x, L_y)$  represents pixel coordinate and  $d$  represents the pixel's distance to camera optical center. The function  $\{f : L_y \rightarrow d\}$  can be obtained (shown in Figure 13).

There is a pitch angle between camera optical axis and the middle line of the measurement bar ( $21.5^\circ$  in this paper); therefore, pixels in the middle line of the measurement bar have different spatial resolutions (mm/pixel), and the bigger the vertical coordinate is, the higher the spatial resolution is. Suppose that  $(L_x, L_y)$  represents pixel coordinate, and  $sr$  represents the pixel's spatial resolution, the function  $\{g : L_y \rightarrow sr\}$  can be obtained (shown in Figure 14).

In the continuous casting steelmaking process, the range of the TCF thickness is 5 mm~50 mm. Given the prior knowledge, the functions  $f$  and  $g$ , and the distance from the TCF upper surface to the optical center, the image region of interest (ROI) on the measurement bar can be obtained. And in the ROI, the Gaussian-Laplace operator is used to detect the gray gradient (shown in Figure 15). While the gray gradient caused by TCF lower surface is detected, distance from TCF lower surface to camera optical center can be obtained by using function  $f$ .

**2.5. Implementation.** The steps below outline the procedure for measuring the thickness of TCF, and the Steps (2)–(7) are circularly implemented every 30 minutes.

(1) *Initialization.* Prepare the mechanical device, move it to the specified location, check the status of the movement bar and the camera, and ensure that they are steadily fixed. Then control the mechanical device to insert the movement bar in the molten steel.

(2) *Thermal Balance.* Keep the movement bar in the molten steel long enough to make it in the thermal balance (30 minutes in this paper). And save the current position as  $p_1$  and current image as  $I_1$ .

(3) *Pulling out the Measurement Bar.* Control the mechanical device to pull out the measurement bar, and save the current position as  $p_2$  and current image as  $I_2$ .

(4) *Distance Calculation from TCF Upper Surface to Camera Optical Center.* Use formula (4) to calculate the distance from TCF upper surface to camera optical center, and save it as  $L_{\text{upper}}$ .

(5) *Distance Calculation from TCF Lower Surface to Camera Optical Center.* Use the Gaussian-Laplace operator to obtain the image gray gradient caused by TCF lower surface, and use the function  $\{f : L_y \rightarrow d\}$  to calculate the distance from TCF lower surface to camera optical center, and save it as  $L_{\text{lower}}$ .

(6) *TCF Thickness Calculation.* Use formula (2) to calculate the TCF thickness.

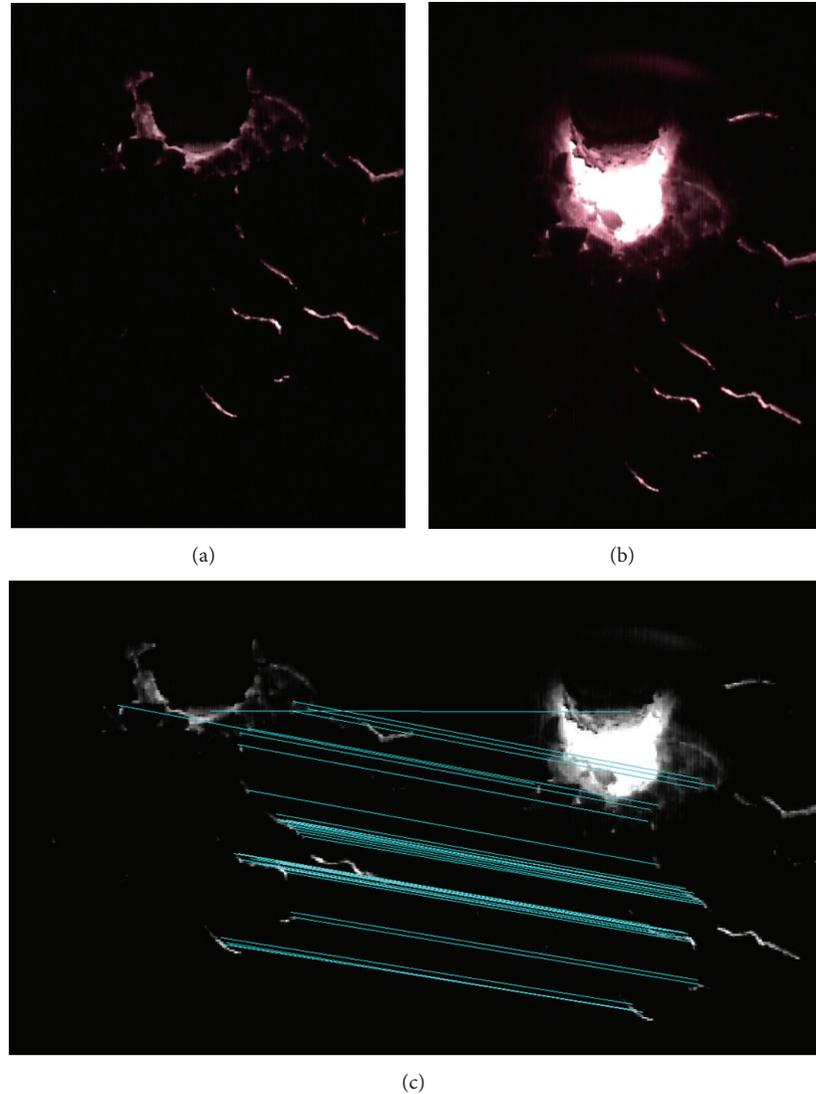


FIGURE 12: Schematic diagram of matching the corresponding pixels in two images based on SIFT descriptors, (a) image  $I_1$  in position  $P_1$ , (b) image  $I_2$  in position  $P_2$  after the measurement bar is pulled out, and (c) matching the corresponding pixels in two images based on SIFT descriptors.

(7) *End Measurement.* Control the mechanical device to insert the measurement bar in the molten steel, and then implement Step (2).

### 3. Experiments and Results

The TCF thickness measurement instrumentation has already been installed and applied in four Chinese steel plants, which are Nanjing Steel Plant located in Jiangsu province since 2010, Daye Steel Plant located in Hubei province since 2011, Hanzhou Steel Plant located in Zhejiang province since 2011, and Sanming Steel Plant located in Fujian province since 2012. The instrumentation works well in these four worksites, and Figure 16 shows the working condition at scene.

*3.1. Experiment of TCF Upper Surface Distance Measurement.* We did this experiment to verify the correctness and accuracy of the method discussed in Section 2.3. The experiment steps are as follows

- (1) We localized the mechanical device in position  $P_1$  and manually measured the distance from the camera to the TCF upper surface (700 mm in this experiment) and then saved the image in this position as  $I_1$ .
- (2) We lifted the mechanical device in position  $P_2$  ( $P_2 - P_1 = 50$  mm in this experiment) and then saved the image in this position as  $I_2$ .
- (3) We did image registration for images  $I_1$  and  $I_2$  based on SIFT and matched the corresponding pixels in

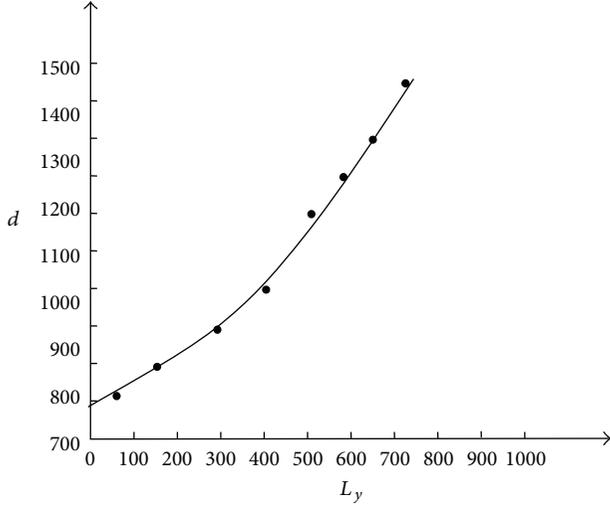


FIGURE 13: Schematic diagram of the relationship between pixels' vertical coordinates in the middle line of the measurement bar and its distance to the camera optical center. • represents the sampling points.

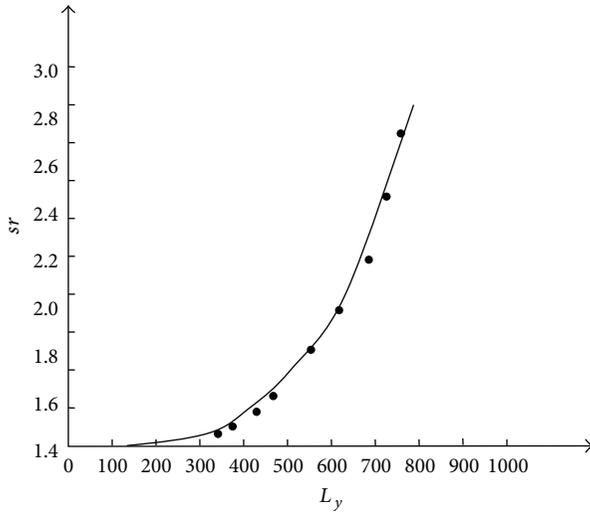


FIGURE 14: Schematic diagram of the relationship between pixels' vertical coordinates in the middle line of the measurement bar and its spatial resolution. • represents the sampling points.

these two images (226 pixels were matched in this experiment).

- (4) We set the parameters in formula (4).  $\Delta h$  represents lifted distance from  $P_1$  to  $P_2$  (50 mm in this experiment),  $2\alpha_0$  represents vertical angle of view ( $42.6^\circ$  in this experiment),  $2\beta_0$  represents horizontal angle of view ( $50^\circ$  in this experiment),  $\gamma_0$  represents angle of pitch ( $21.5^\circ$  in this experiment), and  $(p_{x1}, p_{y1})$  and  $(p_{x2}, p_{y2})$  represent the coordinates of corresponding pixels in images  $I_1$  and  $I_2$ . Table 1 showed the horizontal and vertical angles of view of several kinds lens, which had different sizes and focuses.

TABLE 1: Horizontal and vertical angles of view of several kinds lens, which have different sizes and focuses.

Focus (mm)	Horizontal angle of view for 1/3 inch	Vertical angle of view for 1/3 inch	Horizontal angle of view for 1/4 inch	Vertical angle of view for 1/4 inch
2.5 mm	$96.4^\circ$	$86.2^\circ$	$81.9^\circ$	$72.9^\circ$
2.8 mm	$89.9^\circ$	$79.8^\circ$	$75.6^\circ$	$66.8^\circ$
3.6 mm	$75.7^\circ$	$66^\circ$	$62.2^\circ$	$54.3^\circ$
4 mm	$69.9^\circ$	$60.7^\circ$	$57^\circ$	$49.5^\circ$
6 mm	$50^\circ$	$42.6^\circ$	$39.8^\circ$	$34.2^\circ$
8 mm	$38.5^\circ$	$32.6^\circ$	$30.4^\circ$	$26^\circ$
12 mm	$26.2^\circ$	$22.1^\circ$	$20.5^\circ$	$17.5^\circ$
16 mm	$19.8^\circ$	$16.6^\circ$	$15.4^\circ$	$13.2^\circ$
30 mm	$10.6^\circ$	$8.9^\circ$	$8.3^\circ$	$7^\circ$
60 mm	$5.3^\circ$	$4.5^\circ$	$4.1^\circ$	$3.5^\circ$
100 mm	$3.2^\circ$	$2.7^\circ$	$2.5^\circ$	$2.1^\circ$
200 mm	$1.6^\circ$	$1.3^\circ$	$1.2^\circ$	$1.1^\circ$

- (5) We used formula (4) to calculate the distance from camera to TCF upper surface and compared the calculated results with the actual distance (700 mm in this experiment). The calculation and comparison details were shown in Table 2.

From the comparison shown in Table 2, average deviation is 1.2 mm, and the biggest deviation is 2.2 mm; therefore, we can conclude that the method in Section 2.3 based on monocular vision and image registration can accurately calculate the distance from the camera to TCF upper surface.

**3.2. Comparison between Our Method and Single/Double Wire Methods on TCF Thickness Measurement.** In this experiment, the results of our method were compared with the traditional methods, which are single-wire measurement method and double-wire measurement method. Although single/double wire measurement methods have several disadvantages, such as easily affected by the operator, dangerous to the operator, and poor repeatability, they are still the most common used and the most accredited methods in continuous casting steelmaking process. Therefore, we took these two methods as the golden standard to verify the correctness and accuracy of our method.

To improve the accuracy of the single/double wire measurement methods and reduce the human interference from the operators as far as possible, two operators were asked to join the experiment and strictly obey the following rules: (1) standing on the secure spot and staying alert; (2) vertically inserting the wires into the molten steel; (3) keeping the hand steady while the wires are in the molten steel; (4) keeping the wires in the molten steel for five seconds; (5) pulling the wires out of the molten steel as quickly as possible while the time is up. One operator measured the TCF thickness using one-wire measurement method, and the other one measured the TCF thickness using two-wire measurement method. While our instrumentation measured the TCF thickness,

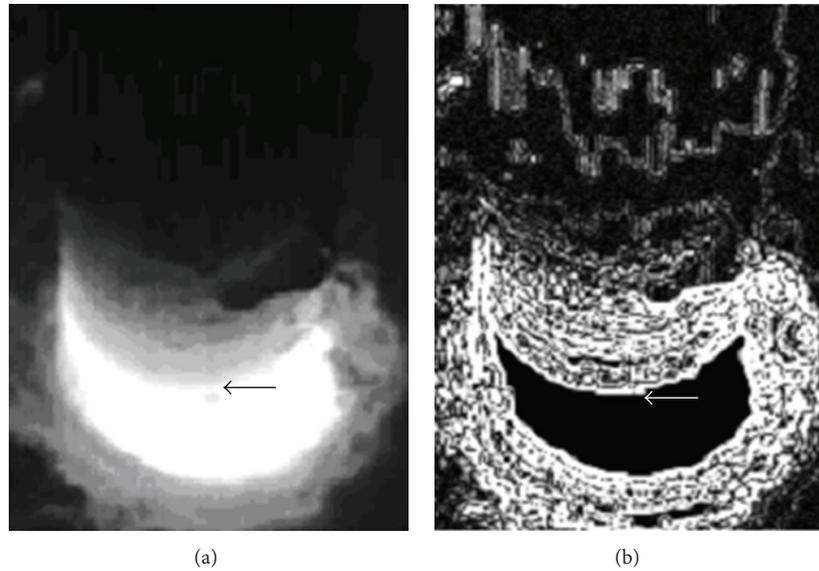


FIGURE 15: (a) Original image, (b) result of gray gradient detection. Arrows represent the gray gradient caused by TCF lower surface.

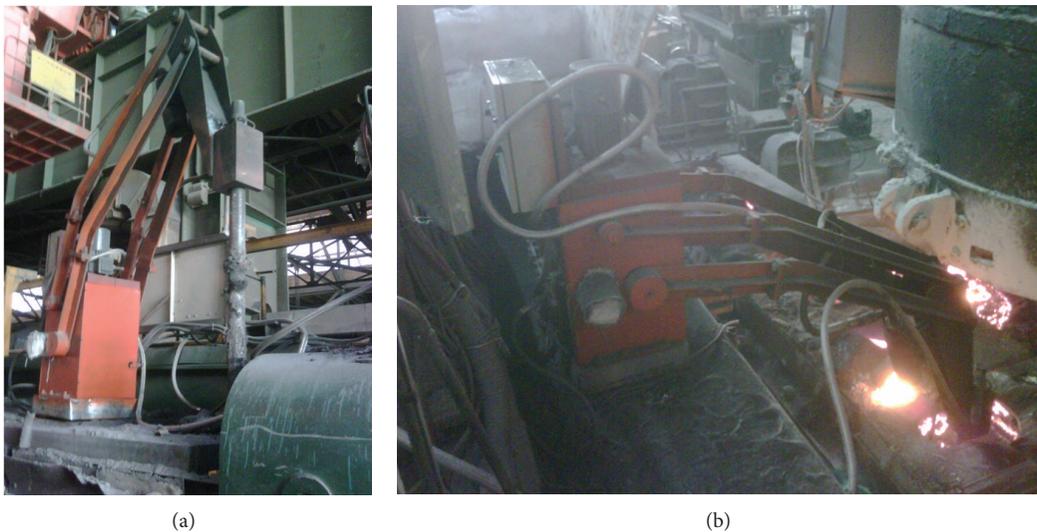


FIGURE 16: Working condition of our instrumentation at scene, (a) in the idle condition, (b) in the working condition.

both operators simultaneously measured the TCF thickness using single/double wire measurement methods. In this way, we can obtain three groups of experimental data at the same time to make the comparison, one from our method, two from single/double wire measurement methods.

The main ingredient of the TCF was shown in Table 3. The operator was asked to add one bag of TCF through each tundish hole every two hours, and the weight of one bag of TCF was about 10 kg.

We did this experiment in two steel plants, respectively, which are Sanming Steel Plant and Nanjing Steel Plant. The error range that can be accepted by the steel plants is  $\pm 5$  mm. Because the variation of TCF thickness is slow, we measured the TCF thickness every 30 minutes using both our method and the traditional single/double wire measurement

methods. The experiments took ten hours in Sanming Steel Plant and nine hours in Nanjing Steel Plant. The measurement results and comparison between our method and the traditional single/double wire measurement methods were shown in Table 4 (Sanming Steel Plant) and Table 5 (Nanjing Steel Plant), and the comparisons were also shown in Figures 17 and 18.

It is shown in Table 4 that in the experiment of Sanming Steel Plant, the biggest deviation of our method from single-wire measurement method is 5.7 mm and the mean deviation of our method from single-wire measurement method is 3.0 mm, and in the experiment of Sanming Steel Plant, the biggest deviation of our method from double-wire measurement method is 6.3 mm and the mean deviation of our method from double-wire measurement method is 2.8 mm.

TABLE 2: Calculated distance from TCF upper surface to camera optical center, and the comparison between calculated distance and actual distance which is manually measured.

Image coordinates before movement	Image coordinates after movement	Calculated distance (mm)	Actual distance (mm)	Deviation (mm)
(455, 220)	(450, 345)	698.6	700	-1.4
(492, 224)	(486, 348)	701.3	700	1.3
(501, 227)	(494, 351)	701.5	700	1.5
(520, 251)	(513, 374)	701.1	700	1.1
(526, 247)	(518, 370)	698.5	700	-1.5
(534, 252)	(526, 374)	700.8	700	0.8
(547, 249)	(539, 372)	700.0	700	0
(551, 248)	(542, 371)	701.8	700	1.8
(330, 277)	(335, 402)	700.9	700	0.9
(474, 239)	(468, 363)	699.1	700	-0.9
(501, 231)	(495, 355)	701.3	700	1.3
(513, 240)	(505, 362)	699.0	700	-1
(520, 246)	(513, 369)	701.4	700	1.4
(528, 254)	(521, 376)	701.8	700	1.8
(529, 260)	(522, 382)	702.2	700	2.2
(535, 255)	(527, 378)	701.1	700	1.1
(535, 259)	(527, 382)	701.0	700	1
(540, 257)	(532, 380)	701.1	700	1.1
(513, 240)	(505, 362)	698.4	700	-1.6
(501, 225)	(494, 349)	700.3	700	0.3

It is shown in Table 5 that in the experiment of Nanjing Steel Plant, the biggest deviation of our method from single-wire measurement method is 6.4 mm and the mean deviation of our method from single-wire measurement method is 2.1 mm, and in the experiment of Nanjing Steel Plant, the biggest deviation of our method from double-wire measurement method is 7.6 mm and the mean deviation of our method from double-wire measurement method is 3.0 mm.

From Tables 4 and 5, we can conclude that our instrumentation and measurement method can accurately measure the TCF thickness, and our method has higher measurement precision and even can replace single/double wire measurement methods in continuous casting steelmaking process.

It is shown in Figures 17 and 18 that these three groups of measured results (our method, single-wire measurement method, and double-wire measurement method) have similar trend. As time went by, the TCF was melted by the molten steel on and on; therefore, the TCF thickness became thinner and thinner until the operator added one more bag of TCF through the tundish hole. The operator added TCF every two hours (at 2, 4, 6, and 8 hours in Figures 17 and 18); therefore, the TCF thickness became much thicker after the TCF was added.

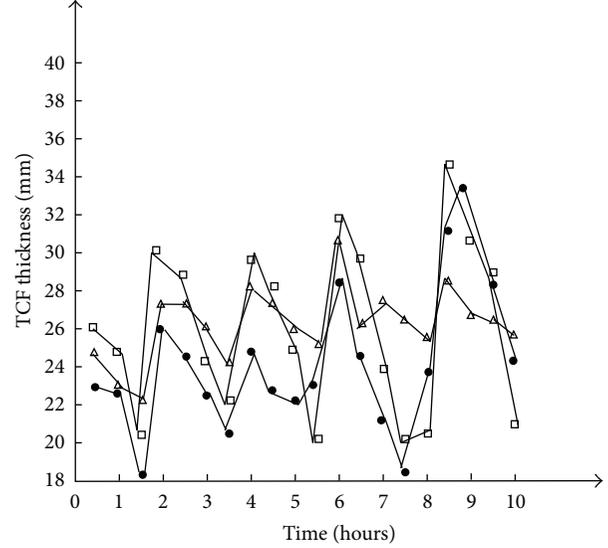


FIGURE 17: Measurement results and comparison between our method and the traditional single/double wire measurement methods. All these data were obtained from Sanming Steel Plant.  $\square$  represents the results of our method,  $\bullet$  represents the results of single-wire measurement method, and  $\triangle$  represents the results of double-wire measurement method.

## 4. Conclusion

This paper presents a novel measurement method and instrumentation to measure the thickness of TCF in continuous casting steelmaking process. The method is based on computer vision algorithms and the instrumentation is specifically designed and built, which was composed of a measurement bar, a mechanical device, a high-definition industrial camera, a PLC, and a computer. Experiments and results showed that our measurement method and instrumentation can accurately measure the thickness of TCF in continuous casting steelmaking process and had good safety for operators. Furthermore, our measurement method and instrumentation can replace traditional single/double wire measurement methods.

## Appendix

The proof of transformation from image coordinates to world coordinates

$$\begin{aligned}
 &\because \text{Point } I \text{ is the joint point of optical} \\
 &\quad \text{center } O \text{ on the plane } ABU \\
 &\therefore \text{Line } OI \perp \text{plane } ABU \\
 &\therefore \text{Plane } OGI \perp \text{plane } ABU \\
 &\because \text{Line } JG \perp \text{line } EF \\
 &\therefore \text{Line } JG \perp \text{plane } OGI.
 \end{aligned} \tag{A.1}$$

Suppose that point  $P$  is any point on the plane  $ABCD$ , whose world coordinate is  $(P_x, P_y)$ . And the corresponding point of

TABLE 3: Main ingredient of the TCF.

Ingredient	SiO <sub>2</sub>	MnO	Al <sub>2</sub> O <sub>3</sub>	CaO	MgO	FeO	Fe <sub>2</sub> O <sub>3</sub>	P <sub>2</sub> O <sub>5</sub>
Content (%)	11.7–15	3.1–4	1–1.6	49–53	3.1–10	11–17	6.9–8	1.7–2

TABLE 4: Measurement results and comparison between our method and the traditional single/double wire measurement methods. All these data were obtained from Sanming Steel Plant.

TCF thickness measured by our method (mm)	TCF thickness measured by single-wire measurement method from operator no. 1 (mm)	TCF thickness measured by double-wire measurement method from operator no. 2 (mm)	Deviation of our method from single-wire measurement method (mm)	Deviation of our method from double-wire measurement method (mm)
26.1	23.9	24.8	2.2	1.3
24.8	23.3	23.5	1.5	1.3
20.7	18.8	22.1	1.9	-1.4
30.5	26.1	27.6	4.4	2.9
29.1	24.6	27.6	4.5	1.5
24.9	22.8	26.1	2.1	-1.2
22.8	21.8	24.1	1	-1.3
29.8	25.1	28.3	4.7	1.5
28.6	22.9	27.3	5.7	1.3
25.7	22.5	26.1	3.2	-0.4
20.5	23.6	25.3	-3.1	-4.8
32.9	28.8	30.8	4.1	2.1
29.8	25.3	26.2	4.5	3.6
24.3	21.8	27.5	2.5	-3.2
20.9	18.5	26.5	2.4	-5.6
20.8	23.9	25.8	-3.1	-5
35.1	31.8	28.8	3.3	6.3
31.6	33.4	26.9	-1.8	4.7
29.1	28.4	26.6	0.7	2.5
21.0	24.4	25.8	-3.4	-4.8
Biggest deviation			5.7	6.3
Mean deviation			3.0	2.8

$P$  in the image is  $P'$ , whose image coordinate is  $(p_x, p_y)$ . As shown in Figure 7, point  $P$  and point  $Q$  have the same vertical coordinate, which is  $GQ$ . Therefore, corresponding point of  $Q$  in the image is  $Q'$ , and  $Q'G' = p_y$ , which shown in Figure 19:

$$\begin{aligned} \operatorname{tg}\alpha &= \frac{G'Q'}{OG'} = \frac{p_y}{G'E'/\operatorname{tg}(\angle G'OE')} \\ &= \frac{p_y}{(1/2)H/\operatorname{tg}\alpha_0} \end{aligned} \quad (\text{A.2})$$

$$\begin{aligned} &= \frac{2p_y \times \operatorname{tg}\alpha_0}{H} \\ \therefore \alpha &= \operatorname{arctg} \frac{2p_y \times \operatorname{tg}\alpha_0}{H}. \end{aligned} \quad (\text{A.3})$$

Similarly,

$$\beta = \operatorname{arctg} \frac{2p_x \times \operatorname{tg}\beta_0}{W}. \quad (\text{A.4})$$

$H$  represents the height of the image:

$$\begin{aligned} OG &= \frac{h}{\cos \gamma_0} \\ \therefore IG &= h \times \operatorname{tg}\gamma_0 \end{aligned} \quad (\text{A.5})$$

$$\begin{aligned} IQ &= h \times \operatorname{tg}(\gamma_0 + \alpha) \\ \therefore GQ &= IQ - IG = h(\operatorname{tg}(\gamma_0 + \alpha) - \operatorname{tg}\gamma_0) \\ \therefore P_y &= GQ = IQ - IG = h(\operatorname{tg}(\gamma_0 + \alpha) - \operatorname{tg}\gamma_0). \end{aligned} \quad (\text{A.6})$$

TABLE 5: Measurement results and comparison between our method and the traditional single/double wire measurement methods. All these data were obtained from the Nanjing Steel Plant.

TCF thickness measured by our method (mm)	TCF thickness measured by single-wire measurement method from operator no. 1 (mm)	TCF thickness measured by double-wire measurement method from operator no. 2 (mm)	Deviation of our method from single-wire measurement method (mm)	Deviation of our method from double-wire measurement method (mm)
33.5	35.8	33.6	-2.3	-0.1
31.0	31.9	30.7	-0.9	0.3
30.0	27.6	29.8	2.4	0.2
37.6	40.3	36.1	-2.7	1.5
35.7	39.5	33.1	-3.8	2.6
31.2	30.0	28.5	1.2	2.7
26.5	29.1	23.8	-2.6	2.7
40.0	40.9	37.6	-0.9	2.4
37.9	31.5	34.8	6.4	3.1
31.0	27.6	29.7	3.4	1.3
21.6	25.9	28.1	-4.3	-6.5
43.8	42.9	38.1	0.9	5.7
38.5	38.7	34.6	-0.2	3.9
28.9	31.6	30.8	-2.7	-1.9
25.3	24.3	28.5	1	-3.2
45.5	45.8	42.5	-0.3	3
40.6	42.6	33.0	-2	7.6
37.1	37.1	32.1	0	5
Biggest deviation			6.4	7.6
Mean deviation			2.1	3.0

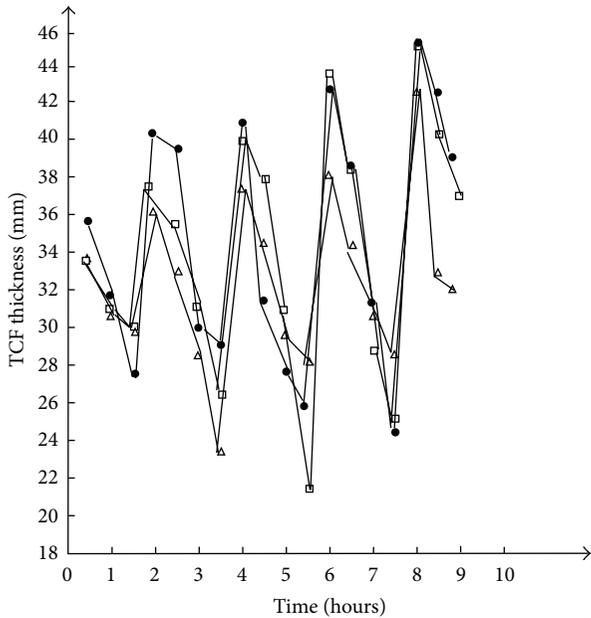


FIGURE 18: Measurement results and comparison between our method and the traditional single/double wire measurement methods. All these data were obtained from Nanjing Steel Plant.  $\square$  represents the results of our method,  $\bullet$  represents the results of single-wire measurement method, and  $\triangle$  represents the results of double-wire measurement method.

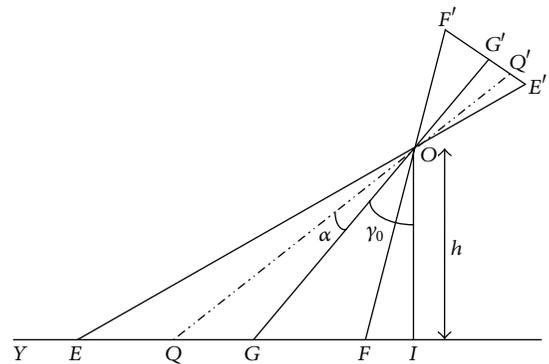


FIGURE 19: Schematic diagram of the corresponding point vertical coordinate between the world coordinates system and the image coordinates system. Point  $Q$  has the same vertical coordinate with the point  $P$  in the plane  $ABCD$ , and the corresponding point of point  $Q$  in the image is point  $Q'$ , and  $Q'G' = p_y$ .

From formula (A.6), the vertical coordinate of point  $P$  in the world coordinates system can be obtained from the image coordinate system:

$$\begin{aligned}
&\therefore IG = h \times tg\gamma_0 \\
&IF = h \times tg(\gamma_0 - \alpha_0) \\
&\therefore GF = IG - IF = h \times (tg\gamma_0 - tg(\gamma_0 - \alpha_0)) \\
&\therefore OG = \frac{h}{\cos\gamma_0} \\
&OF = \frac{h}{\cos(\gamma_0 - \alpha_0)} \\
&\text{and } FC // GJ, JG \perp \text{plane } OGI \\
&\therefore FC \perp \text{plane } OGI \\
&\therefore GJ = OG \times tg\beta_0 = \frac{h}{\cos\gamma_0} \times tg\beta_0 \\
&FC = OF \times tg\beta_0 = \frac{h}{\cos(\gamma_0 - \alpha_0)} \times tg\beta_0 \\
&\therefore \Delta UFC \sim \Delta UGJ \\
&\therefore \frac{FC}{GJ} = \frac{UF}{UG} \\
&\therefore \frac{FC - GJ}{GJ} = \frac{UF - UG}{UG} = \frac{-GF}{UG} \\
&\therefore UG = \frac{GJ \times GF}{GJ - FC} = \frac{h [tg\gamma_0 - tg(\gamma_0 - \alpha_0)] \cos(\gamma_0 - \alpha_0)}{\cos(\gamma_0 - \alpha_0) - \cos\gamma_0} \quad (A.8)
\end{aligned}$$

$$\therefore GL = OG \times tg\beta = \frac{OI}{\cos\gamma_0} \times tg\beta = \frac{h}{\cos\gamma_0} \times tg\beta \quad (A.9)$$

$$\text{In } \Delta UPQ, \frac{GL}{PQ} = \frac{UG}{UQ} \quad (A.10)$$

$$\therefore PQ = \frac{GL(UG + GQ)}{UG}.$$

$GQ$ ,  $UG$ , and  $GL$  can be obtained by formulas (A.6), (A.8), and (A.9), respectively.

Thus,

$$\begin{aligned}
&\therefore P_x = PQ \\
&= \frac{h}{\cos\gamma_0} \times tg\beta \\
&\quad \times \left( 1 + \frac{(tg(\gamma_0 + \alpha) - tg\gamma_0) \times (\cos(\gamma_0 - \alpha_0) - \cos\gamma_0)}{(tg\gamma_0 - tg(\gamma_0 - \alpha_0)) \times \cos(\gamma_0 - \alpha_0)} \right). \quad (A.11)
\end{aligned}$$

From formula (A.11), the horizontal coordinate of point  $P$  in the world coordinates system can be obtained from the image coordinate system.

Therefore, the transformation from the image coordinates system  $(p_x, p_y)$  to the world coordinates system  $(P_x, P_y)$  is

$$\begin{aligned}
P_x &= \frac{h}{\cos\gamma_0} \times tg\beta \\
&\quad \times \left( 1 + \frac{(tg(\gamma_0 + \alpha) - tg\gamma_0) \times (\cos(\gamma_0 - \alpha_0) - \cos\gamma_0)}{(tg\gamma_0 - tg(\gamma_0 - \alpha_0)) \times \cos(\gamma_0 - \alpha_0)} \right), \\
P_y &= h(tg(\gamma_0 + \alpha) - tg\gamma_0), \\
\alpha &= \arctg \frac{2P_y \times tg\alpha_0}{H}, \\
\beta &= \arctg \frac{2P_x \times tg\beta_0}{W}. \quad (A.12)
\end{aligned}$$

## Acknowledgments

This research was supported by the National Natural Science Foundation of China (no. 61101057) and Shenyang Taihe Metallurgical Measurement and Control Ltd.

## References

- [1] H. Kania, K. Nowacki, and T. Lis, "Impact of the density of the mould powder on thickness of the layer of liquid slag in the continuous caster mould," *Metallurgija*, vol. 52, no. 2, pp. 204–206, 2013.
- [2] H. Kania and J. Gawor, "Impact of mould powder density on surface quality and near-surface zone microstructure of cast slab," *Archives of Metallurgy and Materials*, vol. 57, no. 1, pp. 339–345, 2012.
- [3] J. P. Birat, M. Larrecq, and J. Y. Lamant, "The continuous-casting mold—a basic tool for surface quality and strand productivity," in *Proceedings of the 74th Steelmaking Division of the Iron and Steel Society*, pp. 39–50, 1991.
- [4] K.-D. Schmidt, F. Friedel, K.-P. Imlau, W. Jäger, and K. T. Müller, "Consequent improvement of surface quality by systematic analysis of slabs," *Steel Research International*, vol. 74, no. 11–12, pp. 659–666, 2003.
- [5] Z. Hu, Y. Ci, and Z. Xie, "Molten steel level measurement in tundish with heat transfer analysis," *ISIJ International*, vol. 51, no. 10, pp. 1674–1681, 2011.
- [6] Y. Ci, Z. Xie, and H. Zhang, "New approach to continuous temperature measurement of liquid steel in CC tundish," *Journal of Northeastern University*, vol. 25, no. 5, pp. 460–462, 2004.
- [7] H. Zhang and Z. Xie, "Accuracy improvement and structure optimization of continuous temperature measurement sensor for liquid steel," *Acta Metrologica Sinica*, vol. 29, no. 1, pp. 38–41, 2008.
- [8] Z. Xie, Y. Ci, H. Meng, and H. Zhang, "Development of continuous temperature measuring sensor for liquid steel based on blackbody cavity," *Chinese Journal of Scientific Instrument*, vol. 26, no. 5, pp. 446–449, 2005.
- [9] H. Zhang and Z. Xie, "Simulation study on dynamic temperature measurement of liquid steel continuous temperature measurement sensor," *Chinese Journal of Scientific Instrument*, vol. 28, no. 10, pp. 1775–1780, 2007.

- [10] Z. Hua, C. Ying, and X. Zhi, "Dynamic modeling of temperature sensors with great inertia," *Chinese Journal of Scientific Instrument*, vol. 25, no. 4, pp. 261–264, 2004.
- [11] Z. Shumao, M. Guohui, Z. Jiu, and X. Zhi, "Finite element analysis of composite structure continuous temperature-measuring sensor for liquid steel," *Journal of Northeastern University (Natural Science)*, vol. 33, no. 7, pp. 926–929, 2012.
- [12] H. Zhang and Z. Xie, "Static and dynamic uncertainty evaluation of continuous temperature measurement system for liquid steel," *Acta Metrologica Sinica*, vol. 28, no. 4, pp. 329–332, 2007.
- [13] H.-J. Meng, R.-Y. Wu, Z. Xie, and G.-Q. Lin, "Casting speed optimization control system based on continuous temperature measurement of molten steel in CC tundish," *Journal of Northeastern University (Natural Science)*, vol. 26, no. 10, pp. 942–945, 2005.
- [14] S.-M. Zhao, G.-H. Mei, and Z. Xie, "Effect of MgO critical particle size on properties of MgO-C continuous temperature-measuring sensor for liquid steel," *Journal of Iron and Steel Research International*, vol. 18, no. 12, pp. 12–21, 2011.
- [15] W. Liu and Z. Xie, "Design and application of dynamic control system for secondary cooling of billet continuous casting," in *Proceedings of the IEEE International Conference on Advanced Computer Control (ICACC '10)*, pp. 230–233, March 2010.
- [16] B. Wang, Z.-P. Ji, W.-H. Liu, J.-C. Ma, and Z. Xie, "Application of hot strength and ductility test to optimization of secondary cooling system in billet continuous casting process," *Journal of Iron and Steel Research International*, vol. 15, no. 4, pp. 16–20, 2008.
- [17] W.-H. Liu, Z. Xie, Z.-P. Ji, B. Wang, Z.-Y. Lai, and G.-L. Jia, "Dynamic water modeling and application of billet continuous casting," *Journal of Iron and Steel Research International*, vol. 15, no. 2, pp. 14–17, 2008.
- [18] J. Ma, Z. Xie, and G. Jia, "Applying of real-time heat transfer and solidification model on the dynamic control system of billet continuous casting," *ISIJ International*, vol. 48, no. 12, pp. 1722–1727, 2008.
- [19] Z. Ji and Z. Xie, "Multi-objective optimization of continuous casting billet based on ant colony system algorithm," in *Proceedings of the Pacific-Asia Workshop on Computational Intelligence and Industrial Application (PACIIA '08)*, pp. 262–266, December 2008.
- [20] W. Liu, Z. Xie, Z. Ji, and B. Wang, "Research and application of dynamic control system for secondary cooling of billet continuous casting," in *Proceedings of the 2nd IEEE Conference on Industrial Electronics and Applications (ICIEA '07)*, pp. 184–187, May 2007.
- [21] J. Liu, D. Li, Z. Hu, and Z. Xie, "Monocular computer vision image calibration method and its application," in *Proceedings of the IEEE International Conference on Advanced Computer Control (ICACC '10)*, pp. 144–147, March 2010.
- [22] G. Lei, X. Youchun, L. Keqiang, and L. Xiaomin, "Study on real-time distance detection based on monocular vision technique," *Journal of Image and Graphics*, vol. 11, no. 1, pp. 74–81, 2006.
- [23] S. W. Yang, S. A. Scherer, and A. Zell, "An onboard monocular vision system for autonomous takeoff, hovering and landing of a micro aerial vehicle," *Journal of Intelligent & Robotic Systems*, vol. 69, no. 1–4, pp. 499–515, 2013.
- [24] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the 7th IEEE International Conference on Computer Vision (ICCV '99)*, pp. 1150–1157, September 1999.
- [25] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

## Research Article

# Context Prediction of Mobile Users Based on Time-Inferred Pattern Networks: A Probabilistic Approach

Yong-Hyuk Kim<sup>1</sup> and Yourim Yoon<sup>2</sup>

<sup>1</sup> Department of Computer Science & Engineering, Kwangwoon University, 20 Kwangwoon-Ro, Nowon-Gu, Seoul 139-701, Republic of Korea

<sup>2</sup> Future IT R&D Lab., LG Electronics, Umyeon R&D Campus, 38, Baumoe-Ro, Secho-Gu, Seoul 137-724, Republic of Korea

Correspondence should be addressed to Yourim Yoon; [yryoon@soar.snu.ac.kr](mailto:yryoon@soar.snu.ac.kr)

Received 27 May 2013; Accepted 13 July 2013

Academic Editor: Orwa Jaber Housheya

Copyright © 2013 Y.-H. Kim and Y. Yoon. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We present a probabilistic method of predicting context of mobile users based on their historic context data. The presented method predicts general context based on probability theory through a novel graphical data structure, which is a kind of weighted directed multigraphs. User context data are transformed into the new graphical structure, in which each node represents a context or a combined context and each directed edge indicates a context transfer with the time weight inferred from corresponding time data. We also consider the periodic property of context data, and we devise a good solution to context data with such property. Through test, we could show the merits of the presented method.

## 1. Introduction

In recent years, the function of mobile device has been largely extended. Besides call and message service, mobile users usually play games, listen to music, and watch TV, just with their devices. So soon many people cannot get along without their mobile devices [1]. Moreover, mobile device can detect its location and sense the status of the device and current weather such as temperature, humidity, and atmospheric pressure [2]. As such historic context data of mobile users increase cumulatively, now it becomes quite important to provide users with advanced services based on their expected context.

Advertisers want to know mobile users that will do something or locate in some places within a specific time period [3]. Assume that as a result of context prediction a mobile user  $U$  is expected to reach a location  $L$  in near future. Then, as an example service, stores near the location  $L$  can send ads to him or her. In other words, an owner of a restaurant near the location  $L$  can send ads to whom will reach nearby his/her restaurant (see Figure 1(a)). Also, mobile users can get some information in advance about, for example, what to do, where to go, or who to call/message [4]. The user

$U$  may want to get some new information near the location  $L$ . For example, the user  $U$  can get the parking information regarding a resort, where he/she will go probably soon, while staying at the restaurant (see Figure 1(b)). Context prediction of mobile users can provide advertisers and mobile users with useful information services.

For the advanced services, we should predict user's context correctly using his/her previous historic context data. Any context data gathered by mobile device or its sensors are available. Then user contexts are composed of user activities, user locations, weather information, and so on. All of them are time-stamped. In this paper, we present a probabilistic method providing context prediction based on user historic context data. In recent years, there have been many studies to predict context of mobile users. However most of them focused only on location prediction and many of them were based on just machine learning and rule-based techniques. When we compare our method with them, the most distinguishable feature of the proposed method is that it predicts general context based on probability theory through a novel graphical structure with time-based edge weights.

The remainder of the paper is organized as follows. In Section 2, we introduce recent related work about context



FIGURE 1: Example illustrations of intelligent services based on historic context data of mobile users. (a) An example service for advertisers (Ad of a restaurant in the above small box). (b) An example service for users (Parking info of a resort in the above small box).

prediction in mobile environments. In Section 3, we present our prediction scenarios. We propose our probabilistic method using a new graphical structure called *time-inferred pattern network* in Section 4. In Section 5, we provide some test results. Finally we give our conclusions in Section 6.

## 2. Related Work

In recent years, there have been many studies to predict context of mobile users. However most of them focused only on location prediction and many of them were based on just machine learning techniques [5–7]. Also, some of them were based on simple rule-based methods [8, 9].

In this study, we need user’s historic context data, but there have been approaches without user’s historic data. Karmouch and Samaan [10] predicted the traveling trajectory and destination using knowledge of user’s preferences and analyzed spatial information without user’s historic data. Ying et al. [11] predicted the next location of a user’s movement based on both the geographic and semantic features of users’ trajectories. Their prediction model is based on a cluster-based strategy which evaluates the next location of a mobile user based on the frequent behaviors of similar users in the same cluster. Voigtmann et al. [12] also presented a collaborative context prediction technique to overcome the gap of missing context information in the user’s context history.

As similar approaches to ours, there have been studies that are probability based, graph based, or using time concept. Liu and Karimi [7] proposed trajectory prediction methods by a probability-based model and a learning-based model. Wang and Cheng [13] devised an approach for mining periodic maximal promising movement patterns based on a graph structure and a random sampling technique. Chen et al. [14] introduced graph-matching algorithms for mining user movement behavior patterns. Laasonen [15] used other context variables such as time to predict mobile user routes. Bradley and Rashad [16] introduced a time-based location prediction technique by mining mobile sequential patterns. However, all of them are quite simple approaches. Here we present a unified approach using probability theory on a new time-inferred graph structure.

Predicting the location of mobile users was a frequently tackled subtask of mobile context prediction in recent researches. There have also been a few studies for predicting

general context beyond location, but many of them were also based on just machine learning techniques from user behavior [17, 18]. Tseng and Lin [19] mined and predicted user behavior patterns based on the assumption that the location and the service are inherently coexistent. Hassani and Seidl [20] introduced a method for predicting a next health context of mobile users using multiple contextual streams that influence the health context. In this study, we also consider multiple contexts for predicting any type of context.

## 3. Prediction Scenarios

**3.1. Used Context Types.** In mobile computing, location is usually used to approximate context and to implement context-aware applications [6, 7, 9]. However there is more to context than location as illustrated in Figure 2 [21]. We used two types of context data: sensing data and log data. Sensing data consist of location, device status (e.g., idle or calling), activity (e.g., exercising, listening to music, watching TV, or playing games), and weather such as weather type, temperature, humidity, and wind direction/speed. Used log data are as follows: call log, SMS log, played music log, Email log, chatting log, and visited Web log. The above data of each mobile user is periodically transferred to a server. The server analyzes the historic context data and predicts the next context of each user. The next subsections give our prediction scenarios.

**3.2. Prediction on Multiple Context.** From multiple types of historic context data, we predict the context that belongs to the given target context type and will occur the most probably within the given time period. The motivation of using multiple types of context data is from the assumption that heterogeneous context as well as homogeneous context probably affects the next context considerably. Figure 3 illustrates the context prediction based on multiple types of context data. In the figure, given context data of location and weather for three days in time order, we want to predict the third context of the fourth day.

**3.3. Prediction on Periodic Context.** This prediction scenario is an extension of that given in Section 3.2. We assume that users may show different patterns according to some period.

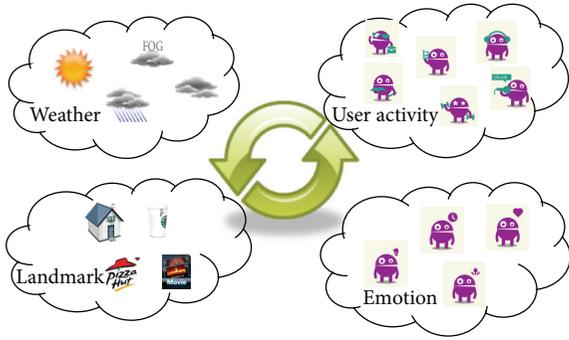


FIGURE 2: An example illustration of various context types.

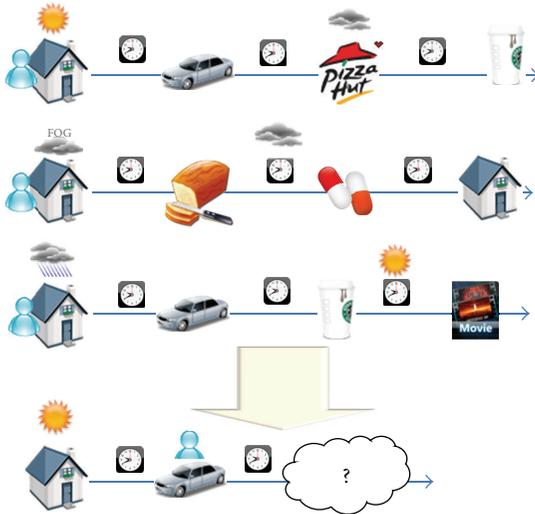


FIGURE 3: An example illustration of context prediction based on multiple types of context data.

For example, the patterns of a typical user on weekday will be quite different from those on weekend. Figure 4 illustrates the context prediction based on periodic context data classified by “a day of the week.” This user may show different patterns on Monday and Saturday. If we ignore this data type of “a day of the week,” even with a good prediction algorithm, inaccurate prediction results may be obtained.

## 4. Proposed Method

**4.1. Input Context Data.** Input data are given in the following form. First, the target context type  $T$  to predict is given. The maximum prediction time is also provided. Historic context data of a user are given in a series of  $\langle$ starting time, ending time, context type, context element $\rangle$ , which are sorted by starting time in nondecreasing order. An example of historic context data can be shown as follows:  $\langle$ 12:00, 13:00, location, restaurant $\rangle$ ,  $\langle$ 13:10, 13:30, activity, listening to music $\rangle$ ,  $\langle$ 13:20, 17:50, weather, rainy $\rangle$ ,  $\langle$ 13:20, 18:00, location, resort $\rangle$ , and so on.

**4.2. Time-Inferred Pattern Network.** Context history can be represented as a graph. For context prediction, we use a new

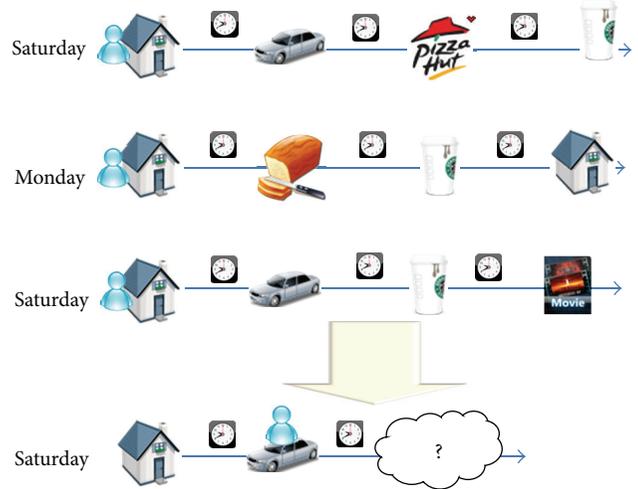


FIGURE 4: An example illustration of context prediction based on periodic context data classified by days of the week.

data structure called *time-inferred pattern network (TIPN)*. We can consider TIPN as a kind of directed weighted multi-graph. Each node means context or its combination. There are three types of nodes. Type-I nodes represent given context elements (e.g., restaurant, listening to music, rainy, resort). Each type-I node has the information of the average and the standard deviation of stay times, where stay time means the difference between starting time and ending time (i.e., stay time = ending time – starting time). Type-II nodes deal with ordered pairs of context elements belonging to different context types (e.g., (restaurant, listening to music), (restaurant, rainy), (listening to music, rainy), (listening to music, resort), (rainy, resort)). Each type-II node has the information of the average, the standard deviation, and the frequency of time gaps, where time gap means the difference between starting times of the two context elements (i.e., time gap = starting time of the second context element – that of the first). Each type-III node is for a series of context elements that are frequently appeared. Here the number of context elements of a type-III node is not limited. Each type-III node has the same information as type-II node, but time gap becomes the difference between starting times of the first context element and the last one. Each directed edge contains movement information from some (combined) context to a target context element. The information contains the average, the standard deviation, and the frequency of time gaps, where time gap is the difference between starting times of starting node and ending one. Each edge also has its weight. The edge weight presents the magnitude of possibility from context to context. The same patterns increase the weight.

**4.2.1. Information of Nodes and Edges.** In this subsection, we note some information of nodes and edges to be used in TIPN. First, we assume that there are  $k$  context types. We have three types of nodes. Information for each type of nodes are given in the following. For a type-I node  $u$  for each context type, we maintain the average of stay times ( $\mu_u$ ), the standard deviation of stay times ( $\sigma_u$ ), and the frequency ( $c_u$ ). Similarly

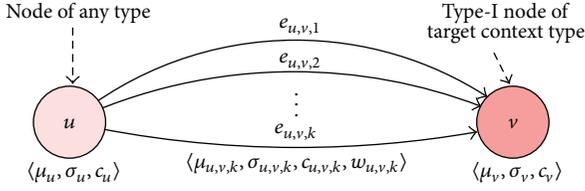


FIGURE 5: Multiple directed edges between nodes in TIPN.

for a type-II node  $u$  for context pair of different context types, we have ordered context pair  $(\gamma_1^u, \gamma_2^u)$ , the average of time gaps  $(\mu_u)$ , the standard deviation of time gaps  $(\sigma_u)$ , and the frequency  $(c_u)$ . Considering a type-III node  $u$  coming from frequently appearing context sequence, we maintain common context sequence  $(\gamma_1^u, \gamma_2^u, \dots, \gamma_{n_u}^u)$ , the average of time gaps  $(\mu_u)$ , the standard deviation of time gaps  $(\sigma_u)$ , and the frequency  $(c_u)$ . Finally, we consider multiple edges between context type  $S$  and target context type  $T$ . Edge set  $E = \bigcup_{u \in S, v \in T} E_{u,v}$ , where  $E_{u,v} = \{e_{u,v,1}, e_{u,v,2}, \dots, e_{u,v,k}, \dots\}$ . The following is information about the  $k$ th edge between node  $u$  and node  $v$ : the average of time gaps  $(\mu_{u,v,k})$ , the standard deviation of time gaps  $(\sigma_{u,v,k})$ , the frequency  $(c_{u,v,k})$ , and the weight  $(w_{u,v,k})$ . Figure 5 shows multiple directed edges between nodes in TIPN.

**4.2.2. Preprocessing.** To make type-III nodes, we find frequent subpatterns from context log. For each context element  $\tau$  in target context type  $T$ , we extract subsequences ending with  $\tau$  within time window. (Since the maximum prediction time is given, we do not use the context data beyond the maximum prediction time when we generate a TIPN.) Let  $n_\tau$  be the number of extracted subsequences and let  $\gamma_1^\tau, \gamma_2^\tau, \dots, \gamma_{n_\tau}^\tau$  be the extracted subsequences. For each pair of subsequences, we find its longest common subsequence (LCS) [22]. Then we have  $n_\tau(n_\tau + 1)/2$  LCSes. Using Levenshtein distance [23] and  $k$ -means clustering [24] upon them, we obtain the optimal  $k$  and  $k$  clusters. For each cluster, we find a central sequence so that its length is as short as possible. We add these central sequences to the type-III node set. If the length of a central sequence is just one, we discard the corresponding node.

**4.2.3. Construction.** We construct a TIPN after the preprocessing of creating type-III nodes. Starting with an empty TIPN composed of isolated nodes, we iteratively construct nodes and edges as reading input context data one by one. Algorithm 1 shows the detailed pseudocode for constructing a TIPN.

When we update each edge, we test a normal distribution against accumulated time-gaps for the edge. In the case that they do not follow normal distributions, we divide the edge into two or more edges. So there may be multiple edges between each pair of nodes. The weight  $w_{u,v,k}$  of each edge  $e_{u,v,k}$  is determined in proportion to its frequency  $f_{u,v,k}$  and the magnitudes of its corresponding time gaps.

**4.2.4. Maintenance.** It is clear that recent context data is more important than old ones. So we devise an aging strategy of

edges to give more weights to recent data than old ones. We periodically apply it to only the edge weights per a unit time, for example, a day, a week, a month, or a year. That is, all edge weights are periodically updated as follows:  $w_{u,v,k} = \alpha \times w_{u,v,k}$  for all edge  $e_{u,v,k}$ , where  $\alpha < 1$ . (We set  $\alpha$  to be 0.9 in our experiments.) Also, we remove weakly connected edges. Given threshold  $t$  for removing edges, we remove edge  $e_{u,v,k}$  when its weight  $w_{u,v,k}$  is less than  $t$ .

In the case that a TIPN has already been constructed and we have added context data, we do the same procedure of Algorithm 1 starting with the previously constructed TIPN instead of an empty TIPN.

### 4.3. Prediction Algorithm

**4.3.1. Assumptions.** In general, we assume that time-gap data for each edge and stay-time data for each node follow a normal distribution. If the collected data is sufficient, our assumption is reasonable by *the law of large numbers* in probability theory [25]. However, in the case that the collected data is sparse, the assumption is not good. In such cases, we assume the data follow a uniform distribution. In more detail, we assume that the data follow  $U[\mu - \sqrt{3}\sigma, \mu + \sqrt{3}\sigma]$ , where  $\mu$  and  $\sigma$  mean the average and the standard deviations of sampled time data, respectively. The following fact supports our assumption.

*Fact 1.* The average and the variance of  $U[\mu - \sqrt{3}\sigma, \mu + \sqrt{3}\sigma]$  are  $\mu$  and  $\sigma^2$ , respectively.

*Proof.* Let a random variable  $X$  follow  $U[\mu - \sqrt{3}\sigma, \mu + \sqrt{3}\sigma]$ , and let  $f(t)$  be the probability density function of  $X$ . Then,

$$f(t) = \begin{cases} \frac{1}{2\sqrt{3}\sigma} & \text{if } \mu - \sqrt{3}\sigma \leq t \leq \mu + \sqrt{3}\sigma \\ 0 & \text{otherwise,} \end{cases}$$

$$\begin{aligned} E(X) &= \int_{-\infty}^{\infty} t f(t) dt \\ &= \int_{\mu - \sqrt{3}\sigma}^{\mu + \sqrt{3}\sigma} \frac{t}{2\sqrt{3}\sigma} dt \\ &= \frac{(\mu + \sqrt{3}\sigma)^2 - (\mu - \sqrt{3}\sigma)^2}{4\sqrt{3}\sigma} \\ &= \mu, \end{aligned}$$

$$\begin{aligned} \text{Var}(X) &= E(X^2) - E(X)^2 \\ &= \int_{-\infty}^{\infty} t^2 f(t) dt - \mu^2 \\ &= \int_{\mu - \sqrt{3}\sigma}^{\mu + \sqrt{3}\sigma} \frac{t^2}{2\sqrt{3}\sigma} dt - \mu^2 \\ &= \frac{(\mu + \sqrt{3}\sigma)^3 - (\mu - \sqrt{3}\sigma)^3}{6\sqrt{3}\sigma} - \mu^2 \\ &= \sigma^2. \end{aligned}$$

(1)  
□

4.3.2. *Sum of Edge Distribution and Node One.* First we consider the sum of normal distributions. It is well known in probability theory, for example, [25]. If  $N_1$  and  $N_2$  are independent random variables that are normally distributed, then their sum is also normally distributed. That is, if  $N_1 \sim N(\mu_1, \sigma_1^2)$ ,  $N_2 \sim N(\mu_2, \sigma_2^2)$ , and  $N_1$  and  $N_2$  are independent, then  $N_1 + N_2 \sim N(\mu_1 + \mu_2, \sigma_1^2 + \sigma_2^2)$ . This means that the sum of two independent normally distributed random variables is normal, with its mean being the sum of the two means, and its variance being the sum of the two variances (i.e., the square of the standard deviation is the sum of the squares of the standard deviations).

Next we consider the sum of uniform distributions. We assume that  $U_1$  and  $U_2$  are independent random variables that are uniformly distributed, that is,  $U_1 \sim U[l_1, u_1]$  and  $U_2 \sim U[l_2, u_2]$ . Let  $U_1^*$  be the uniform random variable obtained by shifting the ranges of  $U_1$  by  $-l_1$ ; that is,  $U_1^* = U_1 - l_1 \sim U[0, u_1 - l_1]$ . Let  $U_2^*$  be the uniform random variable obtained by shifting the ranges of  $U_2$  by  $-l_2$ ; that is,  $U_2^* = U_2 - l_2 \sim U[0, u_2 - l_2]$ . Assuming that  $u_1 - l_1 \leq u_2 - l_2$  without loss of generality, the sum of  $U_1^*$  and  $U_2^*$  is  $U_1 + U_2 - (l_1 + l_2)$ . Then,  $U_1 + U_2 = U_1^* + U_2^* + (l_1 + l_2)$ . The cumulative distribution function of  $U_1^* + U_2^*$  is

$$\begin{aligned}
 P(U_1^* + U_2^* \leq t) &= \int_0^t P(U_1^* = x) P(U_2^* \leq t - x) dx \\
 &= \int_0^t \frac{P(U_2^* \leq t - x)}{u_1 - l_1} dx,
 \end{aligned} \tag{2}$$

where  $t > 0$ . Then,

$$P(U_1^* + U_2^* \leq t) = \begin{cases} 0 & \text{if } t \leq 0 \\ \int_0^t \frac{t-x}{(u_1-l_1)(u_2-l_2)} dx & \text{if } 0 < t \leq u_1-l_1 \\ \int_0^{u_1-l_1} \frac{t-x}{(u_1-l_1)(u_2-l_2)} dx & \text{if } u_1-l_1 < t \leq u_2-l_2 \\ \int_0^{t-(u_2-l_2)} \frac{1}{u_1-l_1} dx + \int_{t-(u_2-l_2)}^{u_1-l_1} \frac{t-x}{(u_1-l_1)(u_2-l_2)} dx & \text{if } u_2-l_2 < t \leq (u_1+u_2)-(l_1+l_2) \\ \int_0^{u_1-l_1} \frac{1}{u_1-l_1} dx & \text{if } t > (u_1+u_2)-(l_1+l_2), \end{cases}$$

$$P(U_1^* + U_2^* \leq t) = \begin{cases} 0 & \text{if } t \leq 0 \\ \frac{t^2}{2(u_1-l_1)(u_2-l_2)} & \text{if } 0 < t \leq u_1-l_1 \\ \frac{2t-(u_1-l_1)}{2(u_2-l_2)} & \text{if } u_1-l_1 < t \leq u_2-l_2 \\ (-t^2 + 2((u_1-l_1) + (u_2-l_2))t - (u_1-l_1)^2 - (u_2-l_2)^2) \times (2(u_1-l_1)(u_2-l_2))^{-1} & \text{if } u_2-l_2 < t \leq (u_1+u_2)-(l_1+l_2) \\ 1 & \text{if } t > (u_1+u_2)-(l_1+l_2). \end{cases} \tag{3}$$

By differentiating the function by  $t$  and then shifting the range of  $U_1^* + U_2^*$  by  $l_1 + l_2$  to the right, we obtain the following probability density function of  $U_1 + U_2$  (see Figure 6):

$$\frac{d}{dt} P(U_1 + U_2 \leq t) = \begin{cases} 0 & \text{if } t \leq l_1 + l_2 \\ \frac{t - (l_1 + l_2)}{(u_1 - l_1)(u_2 - l_2)} & \text{if } l_1 + l_2 < t \leq u_1 + l_2 \\ \frac{1}{u_2 - l_2} & \text{if } u_1 + l_2 < t \leq u_2 + l_1 \\ \frac{(u_1 + u_2) - t}{(u_1 - l_1)(u_2 - l_2)} & \text{if } u_2 + l_1 < t \leq u_1 + u_2 \\ 0 & \text{if } t > u_1 + u_2. \end{cases} \tag{4}$$

Finally we consider the sum of normal distribution and uniform one. Let  $N$  be a random variable that is normally distributed; that is,  $N \sim N(\mu, \sigma^2)$ . Let  $U$  be a random variable that is uniformly distributed; that is,  $U \sim U[l, u]$ . We assume that  $N$  and  $U$  are independent random variables. We cannot obtain the closed form of the probability density function of  $N + U$ , but fortunately the following cumulative distribution function is available:

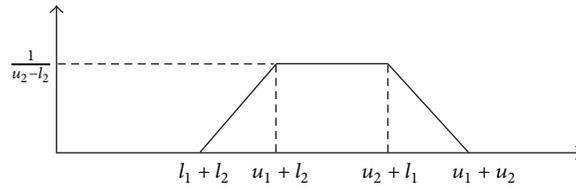
$$\begin{aligned}
 P(U + N \leq t) &= \int_{-\infty}^{\infty} P(N = x) P(U \leq t - x) dx \\
 &= \int_{-\infty}^{\infty} f(x) P(U \leq t - x) dx \\
 &= \int_{-\infty}^{t-u} f(x) dx + \int_{t-u}^{t-l} f(x) \frac{t-x-l}{u-l} dx + \int_{t-l}^{\infty} 0 dx
 \end{aligned}$$

```

Start with an empty TIPN composed of isolated nodes;
For  $i \leftarrow 1$  to  $input\ data\ size - 1$  // read input context data
// constructing type-I nodes and their edges
for  $k \leftarrow 1$  to  $i's\ window\ size - 1$ 
    Update info of type-I node  $n_i$ ;
    if type-I node  $n_{i+k}$  belongs to  $T$  then
        Update edge  $(n_i, n_{i+k})$ ;
// constructing type-II nodes and their edges
for  $k \leftarrow 1$  to  $i's\ window\ size - 2$ 
    if context type of  $n_{i+k} \neq$  context type of  $n_i$  then
        for  $t \leftarrow k + 1$  to  $i's\ window\ size - 1$ 
            if type-I node  $n_{i+t}$  belongs to  $T$  then
                Update info of type-II node  $n' = (n_i, n_{i+k})$ ;
                Update edge  $(n', n_{i+t})$ ;
// constructing type-III nodes and their edges
for  $k \leftarrow 1$  to  $i's\ window\ size - 2$ 
    if  $\exists$  a type-III node  $n''$  starting with the node  $n_i$  then
        for  $t \leftarrow k + 1$  to  $i's\ window\ size - 1$ 
            if type-I node  $n_{i+t}$  belongs to  $T$  then
                Update info of type-III node  $n''$ ;
                Update edge  $(n'', n_{i+t})$ ;

```

ALGORITHM 1: Pseudocode for constructing a TIPN after preprocessing.

FIGURE 6: Sum of two uniform distributions  $U[l_1, u_1]$  and  $U[l_2, u_2]$ .

$$\begin{aligned}
&= P\left(z \leq \frac{t-u-\mu}{\sigma}\right) + \frac{t-l-\mu}{u-l} \\
&\quad \times \int_{t-u}^{t-l} f(x) dx - \frac{\sigma^2}{u-l} \int_{t-u}^{t-l} \frac{x-\mu}{\sigma^2} f(x) dx \\
&= P\left(z \leq \frac{t-u-\mu}{\sigma}\right) \cdot \frac{u-t+\mu}{u-l} \\
&\quad + P\left(z \leq \frac{t-l-\mu}{\sigma}\right) \cdot \frac{t-l-\mu}{u-l} \\
&\quad + \frac{\sigma^2}{u-l} \cdot (f(t-l) - f(t-u)),
\end{aligned} \tag{5}$$

where  $f(x) = (1/\sqrt{2\pi} \cdot \sigma)e^{-(x-\mu)^2/2\sigma^2}$ .

**4.3.3. Algorithm.** When the current time is  $t$  and recent context data are given, we are to list all possible context elements and their possibilities of a specific user within the given time period  $[t_s, t_e]$ , where  $t < t_s < t_e$ . To predict context, we use a TIPN that is already constructed. The possibility  $p_v$  of each target node  $v$  is computed by the following three factors: (i) probability from each node

$u$  to the target node  $v$  based on the edge  $e_{u,v,k}$ , (ii) the context predictability of each node  $u$ , and (iii) the degree of importance of the edge  $e_{u,v,k}$  to the target node  $v$ .

Now we define each factor and give the formula to compute  $p_v$ . (i) First, we calculate the probability  $P(u, v, k)$  from each node  $u$  to the target node  $v$  in relation to the edge  $e_{u,v,k}$ . We assume that time-gap data for each edge and stay-time data for each node follow a normal or uniform distribution. To compute  $P(u, v, k)$  precisely, we need to make good use of the conditional probability and normal sum distribution. Let  $f_{u,v,k}$  and  $g_{u,v,k}$  be the probability density functions of edge  $e_{u,v,k}$  and sum of edge  $e_{u,v,k}$  and node  $v$ , respectively. Let  $\alpha_u$  be  $t -$  (the starting time of the latest context related to node  $u$ ). Then  $P(u, v, k)$  becomes  $1 -$  (probability to arrive at node  $v$  after time  $t_e -$  (probability to depart from node  $v$  before time  $t_s$ )). Hence,

$$P(u, v, k) = 1 - \frac{\int_{t_e-t+\alpha_u}^{\infty} f_{u,v,k}}{\int_{\alpha_u}^{\infty} f_{u,v,k}} - \frac{\int_{\alpha_u}^{t_s-t+\alpha_u} g_{u,v,k}}{\int_{\alpha_u}^{\infty} g_{u,v,k}}. \tag{6}$$

(ii) Context predictability means the easiness degree of prediction for target context type. It can be derived from

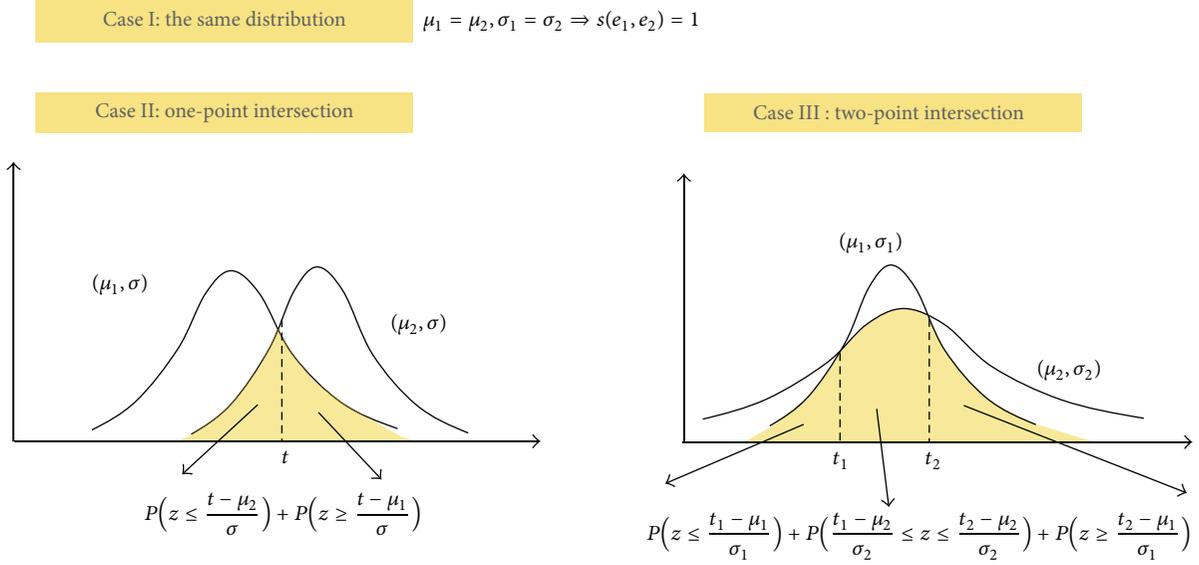


FIGURE 7: Similarity between normal distributions.

a similarity measure between each node  $u$  and target context type  $T$ . We use the variance measure as follows:

$$S(u, T) = |T| \times \text{Var}_{t \in T}(p_{u,t}), \quad \text{where } p_{u,t} = \frac{\sum_k w_{u,t,k}}{\sum_{t \in T, k} w_{u,t,k}}. \quad (7)$$

It ranges from 0 to  $1 - 1/k$ .

(iii) The degree of importance of the edge  $e_{u,v,k}$  from node  $u$  to node  $v$  can be simply defined based on edge weights as follows:

$$D(u, v, k) = \frac{w_{u,v,k}}{\sum_{t \in T, k} w_{u,t,k}}. \quad (8)$$

Now we can define the possibility  $p_v$  of each target node  $v$  as the following weighted sum:

$$p_v = \sum_u \left( S(u, T) \times \sum_k D(u, v, k) \times P(u, v, k) \right). \quad (9)$$

**4.4. Managing Periodic Data.** To manage periodic context data, we define basic periodic elements, for example, the set of “days of the week.” First, we construct a TIPN per each basic periodic element. Some group of basic elements may have similar patterns. If so, we should merge such TIPNs into one before applying the prediction algorithm given in Section 4.3. Similarity  $s(G_A, G_B)$  between any two TIPNs,  $G_A = (N_A, E_A)$  and  $G_B = (N_B, E_B)$ , can be obtained from the normalized summation of similarities  $s(e^A, e^B)$ s between distributions of edges (see the next subsections). That is,

$$s(G_A, G_B) = \sum_i \frac{s(e_i^A, e_i^B)}{|E_A \cup E_B|}. \quad (10)$$

If the similarity between two TIPNs is close to one, we merge the two TIPNs. This process is continued until there is no similar TIPNs. Now we present how to merge two TIPNs. We just merge nodes and edges in the TIPNs. Nodes and edges have the information of their statistical distribution. So we have to consider the sum of distributions. We assume that each node and each edge follow normal or uniform distributions. Since the sum of normal or uniform distributions is given in Section 4.3.2, we have only to obtain its related statistics. Assuming that we have two statistics of the average, the standard deviation, and the frequency, that is,  $(\mu, \sigma, c)$  and  $(\mu', \sigma', c')$ , their sum becomes

$$\left( \frac{c\mu + c'\mu'}{c + c'}, \sqrt{\frac{c(\sigma^2 + \mu^2) + c'(\sigma'^2 + \mu'^2)}{c + c'} - \left(\frac{c\mu + c'\mu'}{c + c'}\right)^2}, c + c' \right). \quad (11)$$

**4.4.1. Similarity between Normal Distributions.** Let edges  $e_1$  and  $e_2$  follow normal distributions,  $N(\mu_1, \sigma_1)$  and  $N(\mu_2, \sigma_2)$ , respectively. As shown in Figure 7, there are three cases to consider. The first case is that  $e_1$  and  $e_2$  follow the same distribution, that is,  $\mu_1 = \mu_2$  and  $\sigma_1 = \sigma_2$ . In this case, it is obvious that  $s(e_1, e_2) = 1$ . The second case is that the probability density functions of  $e_1$  and  $e_2$  have only one intersection. Then  $\mu_1 \neq \mu_2$  and  $\sigma_1 = \sigma_2$ . We assume that  $\mu_1 < \mu_2$  without loss of generality. Since the intersection point  $t$  is  $(\mu_1 + \mu_2)/2$ ,  $s(e_1, e_2) = 2 \cdot P(z \leq (\mu_1 - \mu_2)/2\sigma)$ . The third case is that the probability density functions of  $e_1$  and  $e_2$  have two intersections. In this case,  $\sigma_1 \neq \sigma_2$ . Assume that

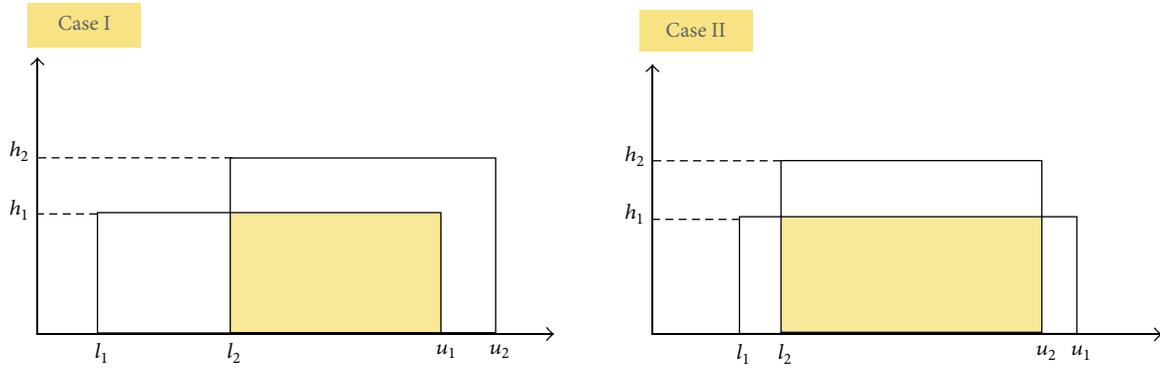


FIGURE 8: Similarity between uniform distributions.

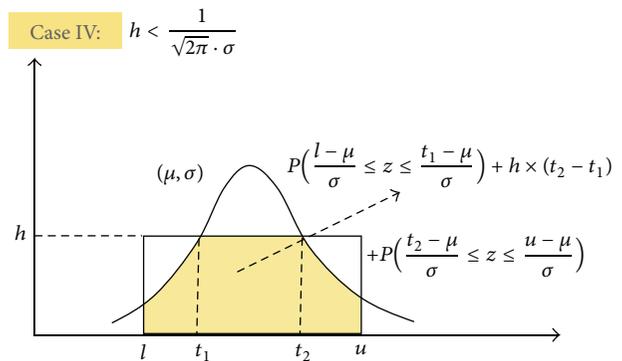
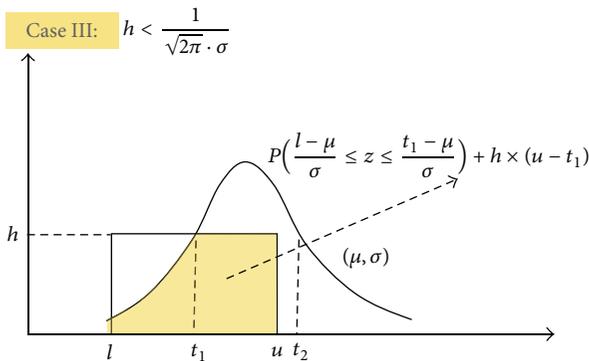
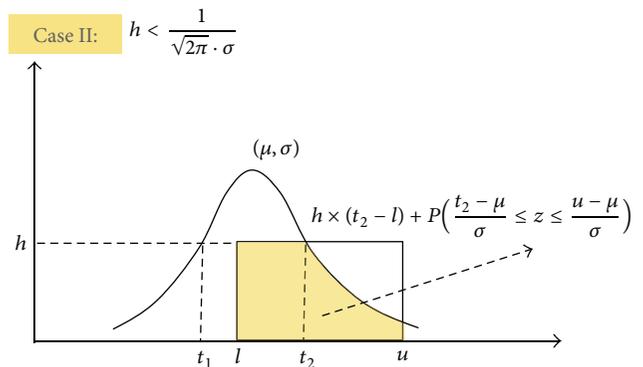
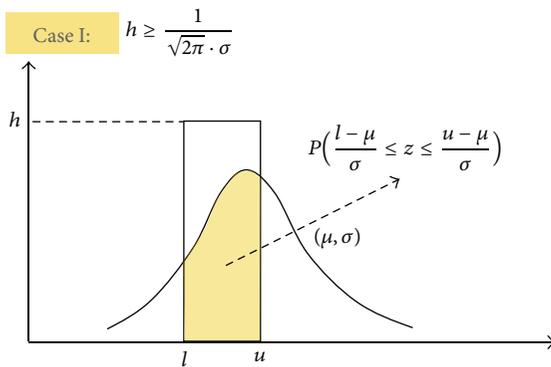


FIGURE 9: Similarity between normal distribution and uniform one.

$\sigma_1 < \sigma_2$  without loss of generality. Then  $s(e_1, e_2) = P(z \leq (t_1 - \mu_1)/\sigma_1) + P((t_1 - \mu_2)/\sigma_2 \leq z \leq (t_2 - \mu_2)/\sigma_2) + P(z \geq (t_2 - \mu_1)/\sigma_1)$ , where  $t_1 = (b - \sqrt{b^2 - ac})/a$ ,  $t_2 = (b + \sqrt{b^2 - ac})/a$ ,  $a = \sigma_2^2 - \sigma_1^2$ ,  $b = \mu_1\sigma_2^2 - \mu_2\sigma_1^2$ , and  $c = \mu_1^2\sigma_2^2 - \mu_2^2\sigma_1^2 + 2\sigma_1^2\sigma_2^2 \ln(\sigma_1/\sigma_2)$ . We can get  $t_1$  and  $t_2$  by solving the equation  $(1/\sqrt{2\pi} \cdot \sigma_1)e^{-(x-\mu_1)^2/2\sigma_1^2} = (1/\sqrt{2\pi} \cdot \sigma_2)e^{-(x-\mu_2)^2/2\sigma_2^2}$ .

4.4.2. *Similarity between Uniform Distributions.* Let edges  $e_1$  and  $e_2$  follow uniform distributions,  $U[l_1, u_1]$  and  $U[l_2, u_2]$ ,

respectively. We assume that  $l_1 \leq l_2$  without loss of generality. As shown in Figure 8, there are two cases to consider. For both cases,  $s(e_1, e_2)$  becomes  $\min(h_1, h_2) \times (\min(u_1, u_2) - \max(l_1, l_2))$ , where  $h_1 = 1/(u_1 - l_1)$  and  $h_2 = 1/(u_2 - l_2)$ .

4.4.3. *Similarity between Normal Distribution and Uniform One.* Let edges  $e_1$  and  $e_2$  follow normal distribution  $N(\mu, \sigma)$  and uniform distribution  $U[l, u]$ , respectively. As shown in Figure 9, there are four cases to consider. We have one case

TABLE 1: Effect of node extension.

Target context type (location)	Probability		
	Type-I nodes	Type-{I, II} nodes	Type-{I, II, III} nodes
L1	<b>0.72</b>	<b>0.85</b>	<b>0.86</b>
L2	0.21	0.11	0.11
L3	0.05	0.03	0.03
L4	0.02	0.01	0.01
L5	0.01	0.00	0.00
Total sum	1.00	1.00	1.00

Values in each column are normalized for their sum to be 1. Location L1 is the answer of our test case.

for  $h \geq 1/\sqrt{2\pi} \cdot \sigma$  and three cases for  $h < 1/\sqrt{2\pi} \cdot \sigma$ , where  $h = 1/(u - l)$ . In the case of  $h \geq 1/\sqrt{2\pi} \cdot \sigma$ ,  $s(e_1, e_2)$  becomes  $P((l - \mu)/\sigma \leq z \leq (u - \mu)/\sigma)$ . When  $h < 1/\sqrt{2\pi} \cdot \sigma$ , the probability density function of  $e_1$  has two intersection points  $t_1$  and  $t_2$  with the line  $y = h$ . Here the points  $t_2$  and  $t_1$ , which are roots of the equation  $h = (1/\sqrt{2\pi} \cdot \sigma)e^{-(x-\mu)^2/2\sigma^2}$ , become  $\mu \pm \sigma\sqrt{-2 \ln(\sqrt{2\pi} \cdot \sigma h)}$ , respectively. The latter three cases can be classified in the following. When  $t_1 \leq l \leq t_2 \leq u$ ,  $s(e_1, e_2) = h \times (t_2 - l) + P((t_2 - \mu)/\sigma \leq z \leq (u - \mu)/\sigma)$ . When  $l \leq t_1 \leq u \leq t_2$ ,  $s(e_1, e_2) = P((l - \mu)/\sigma \leq z \leq (t_1 - \mu)/\sigma) + h \times (u - t_1)$ . When  $l \leq t_1 \leq t_2 \leq u$ ,  $s(e_1, e_2) = P((l - \mu)/\sigma \leq z \leq (t_1 - \mu)/\sigma) + h \times (t_2 - t_1) + P((t_2 - \mu)/\sigma \leq z \leq (u - \mu)/\sigma)$ .

## 5. Test Results

We tested the proposed method using real context data of a mobile user. There were four types of context in our test: location, call, SMS, and activity. They had 5, 6, 2, 2 context elements, respectively. We set location to be target context to predict. The maximum prediction time is 6 hours. Time period to predict is  $t_s = 30$  to  $t_e = 90$  minutes after current time  $t$ . As training data to construct a TIPN, we used data for 160 hours (about 7 days). As test data to apply the prediction algorithm of Section 4.3, we used data for 6 hours before current time  $t$ . The resultant TIPN had 15 type-I nodes, 13 type-II nodes, and 4 type-III nodes. The average number of the context elements of obtained type-III nodes was 2.25. We made two tests. One is the context prediction on nonperiodic weekday data. Table 1 shows the results. To see the effect of node extension (of introducing type-II nodes and type-III ones), we used three TIPNs: TIPN with only type-I nodes, that with type-I nodes and type-II ones, and that with all types of nodes. All the methods successfully predicted context with high probability. We could also see that type-II nodes and type-III nodes help predict context more accurately.

The other test is the context prediction on periodic data. As basic periodic elements, we used days of the week. The test user had two patterns: weekday pattern and weekend one. We made tests for both cases. Table 2 shows the results. "All" means the prediction on the whole context data. "Periodic" means the prediction applying the method of Section 4.4. We could see that "Periodic" was better than "All." In weekend test case, "All" even produced a wrong answer.

TABLE 2: Effect of managing periodic data.

Target context type (location)	Probability			
	Weekday case		Weekend case	
	All	Periodic	All	Periodic
L1	0.18	0.16	0.06	0.04
L2	<b>0.80</b>	<b>0.84</b>	<b>0.49</b>	0.00
L3	0.00	0.00	0.00	0.00
L4	0.01	0.00	0.45	<b>0.96</b>
L5	0.00	0.00	0.00	0.00
Total sum	1.00	1.00	1.00	1.00

Values in each column are normalized for their sum to be 1. Location L2 is the answer of our weekday test case. Location L4 is the answer of our weekend test case.

## 6. Conclusions

We proposed a novel probabilistic approach for context prediction of mobile users based on their historic context data. The proposed method predicts general context based on probability theory through a new graphical structure called *time-inferred pattern network (TIPN)*.

The accumulated context data of a user are transformed into the TIPN, in which each type-I node represents a single context, each type-II node is from a pair of contexts, each type-III node is made from a series of contexts the most generally, and each directed edge indicates a context transfer with the time weight inferred from corresponding time data. With the constructed TIPN, assuming nodes and edges follow normal or uniform distributions, we apply a scoring method derived from probability theory to predict the next context.

Unlike traditional approaches using only location, we used other types of context such as call, SMS, and activity, together with location. We also considered context data with periodic property, providing a good solution for context prediction of mobile users with such patterns. Some empirical studies could show the goodness of the proposed prediction algorithms. Nodes corresponding to more complex context patterns could help predict the next contexts more accurately. So could the classification and clustering of daily context data patterns with periodic property. Although the proposed method showed some merits through a simple comparison from our experiments on small data set, we did not make comparison of the proposed method with other state-of-the-art ones. Such comparison on large-scale data will be a good direction for future work.

## Acknowledgments

The present research has been conducted in 2013 during the sabbatical research year granted by Kwangwoon University. This work was partly supported by the Samsung Electronics Co., Ltd. and the Advanced Research on Meteorological Sciences through the National Institute of Meteorological Research of Korea in 2013 (NIMR-2012-B-1). A preliminary version of this paper appeared in the Proceedings of the

Annual ACM Symposium on Applied Computing, pp. 1015–1019, 2010. The authors would like to thank Mr. Wonkook Kim for his valuable suggestions in improving this paper.

## References

- [1] L. Palen, M. Salzman, and E. Youngs, “Going wireless: behavior & practice of new mobile phone users,” in *Proceedings of the ACM Conference on Computer Supported Cooperative Work (CSCW '00)*, pp. 201–210, Philadelphia, Pa, USA, December 2000.
- [2] D. Siewiorek, A. Smailagic, J. Furukawa et al., “SenSay: a context-aware mobile phone,” in *Proceedings of the 7th IEEE International Symposium on Wearable Computers (ISWC '03)*, pp. 248–249, White Plains, NY, USA, October 2003.
- [3] C. Ververidis and G. C. Polyzos, “Mobile marketing using location based services,” in *Proceedings of the 1st International Conference on Mobile-Business (ICMB '02)*, Athens, Greece, July 2002.
- [4] G. D. Abowd, C. G. Atkeson, J. Hong, S. Long, R. Kooper, and M. Pinkerton, “Cyberguide: a mobile context-aware tour guide,” *Wireless Networks*, vol. 3, no. 5, pp. 421–433, 1997.
- [5] T. Anagnostopoulos, C. Anagnostopoulos, and S. Hadjiefthymiades, “Efficient location prediction in mobile cellular networks,” *International Journal of Wireless Information Networks*, vol. 19, no. 2, pp. 97–111, 2012.
- [6] T. Anagnostopoulos, C. Anagnostopoulos, S. Hadjiefthymiades, M. Kyriakakos, and A. Kalousis, “Predicting the location of mobile users: a machine learning approach,” in *Proceedings of the International Conference on Pervasive Services (ICPS '09)*, pp. 65–72, London, UK, July 2009.
- [7] X. Liu and H. A. Karimi, “Location awareness through trajectory prediction,” *Computers, Environment and Urban Systems*, vol. 30, no. 6, pp. 741–756, 2006.
- [8] T. H. N. Vu, K. H. Ryu, and N. Park, “A method for predicting future location of mobile user for location-based services system,” *Computers & Industrial Engineering*, vol. 57, no. 1, pp. 91–105, 2009.
- [9] G. Yavaş, D. Katsaros, Ö. Ulusoy, and Y. Manolopoulos, “A data mining approach for location prediction in mobile environments,” *Data & Knowledge Engineering*, vol. 54, no. 2, pp. 121–146, 2005.
- [10] A. Karmouch and N. Samaan, “A mobility prediction architecture based on contextual knowledge and spatial conceptual maps,” *IEEE Transactions on Mobile Computing*, vol. 4, no. 6, pp. 537–551, 2005.
- [11] J. J.-C. Ying, W.-C. Lee, T.-C. Weng, and V. S. Tseng, “Semantic trajectory mining for location prediction,” in *Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (GIS '11)*, pp. 34–43, Chicago, Ill, USA, November 2011.
- [12] C. Voigtmann, S. L. Lau, and K. David, “A collaborative context prediction technique,” in *Proceedings of the 73rd IEEE Vehicular Technology Conference (VTC Spring '11)*, pp. 1–5, Yokohama, Japan, May 2011.
- [13] Y.-T. Wang and J.-T. Cheng, “Mining periodic movement patterns of mobile phone users based on an efficient sampling approach,” *Applied Intelligence*, vol. 35, no. 1, pp. 32–40, 2011.
- [14] T.-S. Chen, Y.-S. Chou, and T.-C. Chen, “Mining user movement behavior patterns in a mobile service environment,” *IEEE Transactions on Systems, Man, and Cybernetics A*, vol. 42, no. 1, pp. 87–101, 2012.
- [15] K. Laasonen, “Clustering and prediction of mobile user routes from cellular data,” in *Proceedings of the 9th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD '05)*, pp. 569–576, Porto, Portugal, October 2005.
- [16] J. Bradley and S. Rashad, “Time-based location prediction technique for wireless cellular networks,” in *Emerging Trends in Computing, Informatics, Systems Sciences, and Engineering*, T. Sobh and K. Elleithy, Eds., vol. 151 of *Lecture Notes in Electrical Engineering*, pp. 937–947, 2013.
- [17] S. Lee and K. C. Lee, “Context-prediction performance by a dynamic Bayesian network: emphasis on location prediction in ubiquitous decision support environment,” *Expert Systems with Applications*, vol. 39, no. 5, pp. 4908–4914, 2012.
- [18] R. Mayrhofer, H. Radi, and A. Ferscha, “Recognizing and predicting context by learning from user behavior,” in *Proceedings of the International Conference on Advances in Mobile Multimedia (MoMM '03)*, pp. 25–35, 2003.
- [19] V. S. Tseng and K. W. Lin, “Efficient mining and prediction of user behavior patterns in mobile web systems,” *Information & Software Technology*, vol. 48, no. 6, pp. 357–369, 2006.
- [20] M. Hassani and T. Seidl, “Towards a mobile health context prediction: sequential pattern mining in multiple streams,” in *Proceedings of the 12th IEEE International Conference on Mobile Data Management (MDM '11)*, vol. 2, pp. 55–57, Luleå, Sweden, June 2011.
- [21] A. Schmidt, M. Beigl, and H.-W. Gellersen, “There is more to context than location,” *Computers & Graphics*, vol. 23, no. 6, pp. 893–901, 1999.
- [22] D. S. Hirschberg, “Algorithms for the longest common subsequence problem,” *Journal of the Association for Computing Machinery*, vol. 24, no. 4, pp. 664–675, 1977.
- [23] R. A. Wagner and M. J. Fischer, “The string-to-string correction problem,” *Journal of the Association for Computing Machinery*, vol. 21, no. 1, pp. 168–173, 1974.
- [24] J. A. Hartigan and M. A. Wong, “Algorithm as 136: a  $k$ -means clustering algorithm,” *Journal of the Royal Statistical Society C*, vol. 28, no. 1, pp. 100–108, 1979.
- [25] C. M. Grinstead and J. L. Snell, *Introduction to Probability*, American Mathematical Society, 2nd edition, 1997.

## Research Article

# Emotion Expression of Robot with Personality

Xue Hu,<sup>1,2</sup> Lun Xie,<sup>1,2</sup> Xin Liu,<sup>2,3</sup> and Zhiliang Wang<sup>1,2</sup>

<sup>1</sup> School of Computer and Communication Engineering, University of Science and Technology Beijing, Beijing 100083, China

<sup>2</sup> Beijing Key Laboratory of Knowledge Engineering for Materials Science, Beijing 100083, China

<sup>3</sup> School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing 100083, China

Correspondence should be addressed to Lun Xie; xielun@ustb.edu.cn

Received 19 April 2013; Accepted 8 July 2013

Academic Editor: Vishal Bhatnagar

Copyright © 2013 Xue Hu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A robot emotional expression model based on Hidden Markov Model (HMM) is built to enable robots which have different personalities to response in a more satisfactory emotional level. Gross emotion regulation theory and Five Factors Model (FFM) which are the theoretical basis are firstly described. And then the importance of the personality effect on the emotion expression process is proposed, and how to make the effect quantization is discussed. After that, the algorithm of HMM is used to describe the process of emotional state transition and expression, and the performance transferring probability affected by personality is calculated. At last, the algorithm model is simulated and applied in a robot platform. The results prove that the emotional expression model can acquire humanlike expressions and improve the human-computer interaction.

## 1. Introduction

In the studies of psychology, emotion is a state which is aroused and experienced by the individual constantly. The aroused process can be either conscious or unconscious. The emotional response sometimes is affected by the environment, but sometimes it comes into contradictions and conflicts and is incompatible with a specific situations; thus, it is necessary for the individuals to adapt to the environment using emotional regulation [1]. Emotion regulation is a process that individuals manage and change the generation and expression of emotions by strategies. Thompson thought that “Emotional regulation may be defined as the extrinsic and intrinsic processes responsible for monitoring, evaluating, and modifying emotional reactions, especially their intensive and temporal features” [2]. Gross thought emotion regulation meant the individuals influenced what kind of emotions they would have, when to have them, and how to experiment and express them [3].

With the development of artificial psychology, robotics, and computer technology, robots with different functions gradually change people's work and life and help human

beings to engage in complex and complicated manual labor. In addition to the intelligent features, mental activities like emotion, personality, willing, and creation are necessary to robots. Thus, emotion regulation is important to robots in human-computer interaction. Lazarus and Folkman proposed the way of coping paradigm and distinguished the emotion regulation between problem-focused coping—persons' attempts to change the situation or get rid of menaces caused by tension, typically by using problem-solving strategies—and emotion-focused coping—persons' attempts to lessen emotional distress by engaging behavioral or cognitive regulation strategies [4]. It regarded emotion regulation as a one-off act. Two-stage process model of emotion regulation proposed by Gross thought it expanded in the emotional process, which meant different strategies were used in different stages [1]. All these studies ignored the effect of the individual personality characteristics on the process of emotional generation and expression. Some researchers described the process of emotional state transition by the mathematical model of Finite State Machine (FSM), which included the influence of individual personality and emotional stimulus [5]. Some others applied backfeed loop [6],

B-model [7], nonparametric cumulative sum (CUSUM) [8], and et cetera to changing the cognitive and expression of emotion. However, something about how to quantify the personality characteristic parameter and how the personality influences the emotional expression in the process of emotion regulation have not been studied clearly.

In this paper, we firstly describe Gross' emotion regulation process briefly and analyze the influence of emotional expression strategy over emotional behaviors. Then, based on HMM, we quantify the Gross' process model of emotion regulation and expound the transfer and expression of emotional states. According to the Five Factor Model scales, the effect of personality on expression suppression strategy is calculated in HMM. Finally, the mathematic model of robot with personality is built and applied in the human-computer interaction situation.

## 2. Gross' Process Model of Emotion Regulation and the Five Factor Model

*2.1. Gross Model of Emotion Regulation.* Different strategies are used in different stages of emotion generation. There are two parts of emotional regulation in the Gross' model: antecedent-focused emotion regulation and response-focused emotion regulation. The antecedent-focused emotion regulation is divided into four different types of strategies: situation selection, situation modification, attentional deployment, and cognitive change. And the fifth strategy which is named as response modulation is a response-focused strategy. Figure 1 shows an overview of these strategies.

Situation selection is the first antecedent-focused emotion regulation strategy in Gross' model. It refers to the fact that individuals unconsciously approach or avoid certain events or occasions to regulate their own emotions. The second antecedent-focused emotion regulation strategy in the model is situation modification, which is an attempt to control and adjust uncomfortable emotional events in the environment. The third antecedent-focused emotion regulation strategy is attentional deployment, which means individuals choose one or more attentional focuses in many aspects of the situation. The fourth antecedent-focused emotion regulation strategy is cognitive change that refers to selecting the possible meanings to the emotional events. Different cognitive meaning may generate different emotion states. The fifth emotion regulation strategy, response modulation, a response-focused strategy, happens after emotion arousal and affects emotional responses like psychological experience, behavioral expression, and physiological responses.

According to Gross' model, there are many emotional regulation strategies in the process of emotion generation. The most common and useful strategies are reappraisal and suppression. Reappraisal, which is an antecedent-focused emotion regulation strategy, changes individual's cognitive understanding of emotional events so as to change emotional experience. Suppression is a response-focused strategy and is used after emotional generation. Individuals use this strategy to change tendency of emotional reaction,

mainly including expressions and physiological responses. This strategy mobilizes the individual's self-control, but it only regulates the expression of emotions and external behavior, and the emotional experience has not changed. Individuals can control their emotions better with a stronger capability of self-control. In the same situation, individuals with different personalities may have different emotional behavior. This paper focuses on how suppression strategy influences emotion expression and what is the effect of personality.

*2.2. Model of Personality.* Personality is one's responses of thinking, emotion, and behavior, influenced by heredity and environment. In a period of time, it has the character of stability. One's personality has an effect on the process of emotional generation, reappraisal, and expression [9]. In psychological research, the Five Factor Model (FFM) [10] which is also named as the OCEAN Model is one of the most recent models proposed so far. The FFM contains five basic dimensions: openness, conscientiousness, extraversion, agreeableness, and neuroticism. The trait of openness describes a person's cognitive style. Conscientiousness means the way we control, manage and regulate our own impulse. The extraversion trait represents the number and density of human interaction, the need for stimulation, and the ability obtaining pleasure. Agreeableness examines the individual's attitude to others, and neuroticism reflects individual's process of emotion regulation, the tendency of negative emotion experience, and the instability of emotion. Each trait is represented by one dimension, and thus, the FFM is a five-dimension model. The five-element vector  $\mathbf{P} = [p_1 \ p_2 \ p_3 \ p_4 \ p_5]$  refers to one's personality, and  $p_i$ ,  $i = 1, 2, \dots, 5$  separately stands for the value of openness, conscientiousness, extraversion, agreeableness, and neuroticism. The effect of personality on suppression strategy in Gross' emotion regulation model is mainly discussed in this paper, and the weight of effect under each personality trait is calculated by Analytic Hierarchy Process (AHP).

*2.3. Expression Parameter of Personality Suppression.* Individual's personality traits are evaluated by NEO-PI-R scales, and personalized vector  $\mathbf{P} = [p_1 \ p_2 \ p_3 \ p_4 \ p_5]$  could be obtained. The five factors have different effects on suppression, and AHP is used to calculate the effect weight  $w_1, w_2, w_3, w_4, w_5$ , from which we can obtain weight vector  $\mathbf{W} = [w_1 \ w_2 \ w_3 \ w_4 \ w_5]^T$ .

- (a) Firstly building simplified hierarchical model as shown in Figure 2. This simplified model contains two layers: destination and criteria. Suppression strategy is the destination, and traits in FFM are the criteria.
- (b) Constructing the pairwise comparison matrix  $A$ . The comparative value between trait  $p_i$  and  $p_j$  is shown in Table 1. The bigger the value is, the more influential  $p_i$  is compared with  $p_j$  to the suppression of emotion expression.

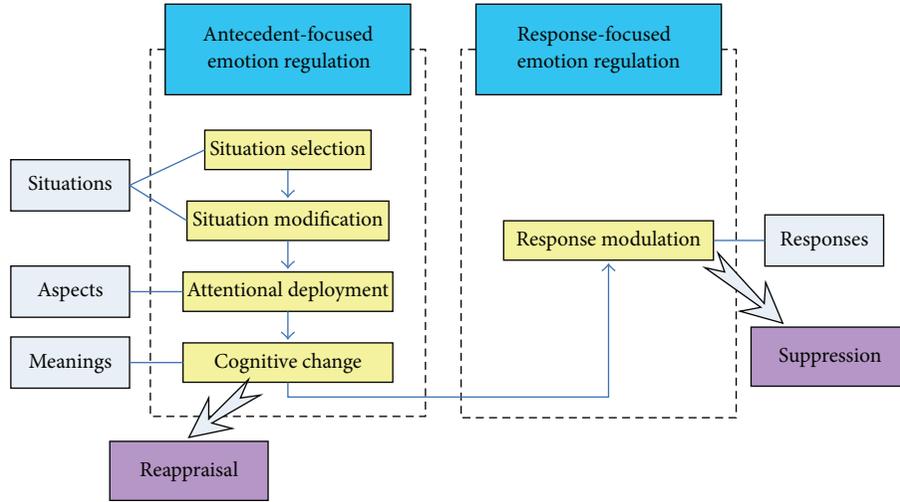


FIGURE 1: Strategies of Gross emotion regulation model.

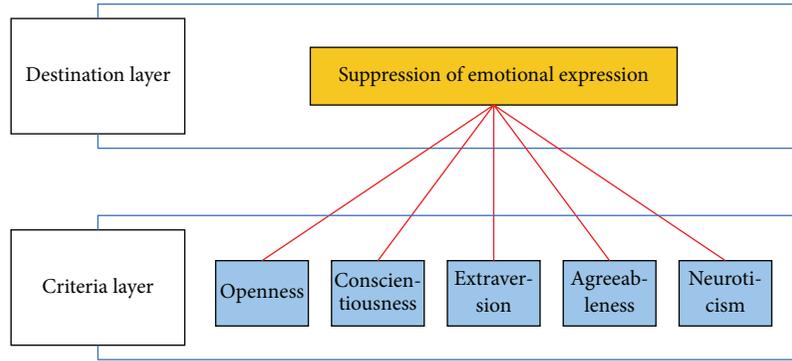


FIGURE 2: The simplified hierarchical model.

Then, the comparison matrix  $A$  can be obtained as follows and  $a_{ji} = 1/a_{ij}$ :

$$\mathbf{A} = (a_{ij})_{5 \times 5} = \begin{pmatrix} 1 & 5 & \frac{1}{5} & 3 & \frac{1}{3} \\ \frac{1}{5} & 1 & \frac{1}{9} & \frac{1}{3} & \frac{1}{7} \\ 5 & 9 & 1 & 7 & 3 \\ \frac{1}{3} & 3 & \frac{1}{7} & 1 & \frac{1}{5} \\ 3 & 7 & \frac{1}{3} & 5 & 1 \end{pmatrix}. \quad (1)$$

Normalizing  $A$  for each column vector by the following formula,  $A'$  is obtained:

$$a'_{ij} = \frac{a_{ij}}{\sum_{i=1}^5 a_{ij}}, \quad (i = 1, 2, \dots, 5, j = 1, 2, \dots, 5), \quad (2)$$

$$\mathbf{A}' = (a'_{ij})_{5 \times 5} = \begin{pmatrix} 0.105 & 0.200 & 0.112 & 0.184 & 0.071 \\ 0.021 & 0.040 & 0.062 & 0.020 & 0.031 \\ 0.524 & 0.360 & 0.560 & 0.429 & 0.642 \\ 0.035 & 0.120 & 0.080 & 0.061 & 0.043 \\ 0.312 & 0.280 & 0.187 & 0.306 & 0.214 \end{pmatrix}. \quad (3)$$

$\bar{w}_i = \sum_{j=1}^5 a'_{ij}$  is calculated by summing  $a'_{ij}$  in row, and  $w_i$  is normalized by the following formula:

$$w_i = \frac{\bar{w}_i}{\sum_{i=1}^5 \bar{w}_i} = \frac{\sum_{j=1}^5 a'_{ij}}{\sum_{i=1}^5 \sum_{j=1}^5 a'_{ij}}. \quad (4)$$

Finally, the weight vector can be gained:  $\mathbf{W} = [0.134 \ 0.035 \ 0.503 \ 0.068 \ 0.260]^T$ .

The ability suppressed by personality in the process of emotional expression is defined as follows:

$$E = \mathbf{P} \times \mathbf{W} = \sum_{i=1}^5 p_i \cdot w_i, \quad (5)$$

$$\eta = \frac{E}{\sum_{i=1}^5 p_i}. \quad (6)$$

$\eta$  is the expression parameter of personality suppression.

TABLE 1: The comparative value between traits.

Suppression of emotion expression	Openness	Conscientiousness	Extraversion	Agreeableness	Neuroticism
Openness	1	5	1/5	3	1/3
Conscientiousness	1/5	1	1/9	1/3	1/7
Extraversion	5	9	1	7	3
Agreeableness	1/3	3	1/7	1	1/5
Neuroticism	3	7	1/3	5	1

### 3. Emotion Regulation Model Based on HMM and the Effect of Personality on Emotion Expression

*3.1. Gross' Emotion Regulation Model Based on HMM.* The emotional process includes generation of emotion and expression of behavior. Emotional arousal is a random process, which is not only in connection with the current stimulus but also the emotional state of the individual. Behavioral expression is a random process of emotional expression after psychological experience generated. The two kinds of random process are in line with HMM, and thus we can use it to describe Gross' process model of emotion regulation in a mathematical method [5, 11].

HMM is a statistical probability model including hidden states and Markov chain that is represented by parameters. As shown in Figure 3, HMM is a double stochastic process that consists of a Markov chain and a general stochastic process.

Markov chain is used to describe the transfer of emotional states by the transition probability. And the general stochastic process is used to describe the relationship between the emotional states and the observed sequence by the observation probability. The process of states transition cannot be observed, thus the model is named as "hidden" Markov model.

HMM consists of two-state sets and three probability matrices.

- Hidden state set  $S = [S_1, S_2, \dots, S_N]$ : these states, that meet the Markov properties, usually cannot be obtained through direct observation.
- Observation state set  $O = [O_1, O_2, \dots, O_M]$ : it is a state set associated with hidden states, and its state number can be different from the hidden state set's.
- Initial state probability matrix  $\pi = [p_1, p_2, \dots, p_N]$ ,  $\sum_{i=1}^N p_i = 1$ : it refers to the probability distribution of hidden state in the initial moment.
- Hidden state transition probability matrix  $\mathbf{A} = (a_{ij})_{N \times N}$ ,  $\sum_{j=1}^N a_{ij} = 1$ : it describes the transition probabilities between the hidden states in HMM.
- Observation state transition probability matrix  $\mathbf{B} = (b_{jk})_{N \times M}$ ,  $\sum_{k=1}^M b_{jk} = 1$ : it represents the probability that when hidden state is  $S_j$ , the observed state is  $O_k$ , here,  $b_{jk} = p(O_k | S_j)$ ,  $1 \leq j \leq N$ ,  $1 \leq k \leq M$ .

Currently, emotion classification models used commonly in human-computer interaction are the OCC model [12]

which specifies 22 categories for emotions, the PAD model [13] that contains three dimensions of pleasure, arousal, and dominance, and the basic emotion model identified by Ekman [14] which includes happiness, anger, disgust, fear, sadness, and surprise. The robot's emotional state set is defined as  $S = [S_1, S_2, \dots, S_N]$ .  $N$  is the number of emotional states, and  $q_i$ ,  $i = 1, 2, \dots, N$  is the probability that state  $S_i$  is generated,  $\sum_{i=1}^N q_i = 1$ ,  $0 \leq q_i \leq 1$  ( $i = 1, 2, \dots, N$ ).

The probability space of emotion state can be regarded as

$$\begin{pmatrix} S \\ Q \end{pmatrix} = \begin{pmatrix} S_1 & S_2 & \cdots & S_N \\ q_1 & q_2 & \cdots & q_N \end{pmatrix}. \quad (7)$$

In this paper, in addition to the six basic emotions, calming is used in the robot emotional states, and this means  $N = 7$ .

Expression of human is various, but most psychologists agree that all expression can be divided into six categories: happiness, anger, disgust, fear, sadness, and surprise, corresponding to the six basic emotions, respectively. The emotion of robot can be observed by the expression of its output, which is the observed state in the observation matrix. Its expression is different to basic emotions because of the different intensity of emotion [15]. The six basic emotions are divided into three area of strength: low, mid, and high, so there are three expressions corresponding to each basic emotion. Therefore, the robot has 19 kinds of expressions including calming, and  $M = 19$ .

The HMM emotion model presented in this paper is about the effect of individual's personality on the process of emotion expression, and the emotional inherent performance which reflects the person's emotional variation, has nothing to do with the character of the individual. In human interaction, we will make a description to the interacted one—such as one person is very cheerful and optimistic, or one person is stable and cautious—obviously it is very difficult for a robot to describe a person with his own characters. In the paper, we assume that the emotion state of a robot has been generated, which means the hidden state has come out by Markov chain, and personality influences the expression of emotion state. Assuming that the robot is in a certain initial state, and the initial probability distribution is  $\pi_0$ , the emotional state probability distribution changes under the hidden state transition probability matrix  $\mathbf{A}$  and generates the hidden state  $S_j$ ; using the suppression strategy, the output of expression  $O_k$  can be obtained by the observation state transition probability matrix  $\mathbf{B}'$  which is corrected by the personality suppression of expression parameter  $\eta$ .

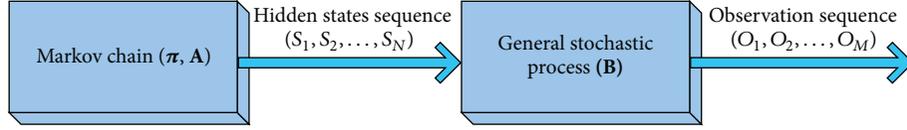
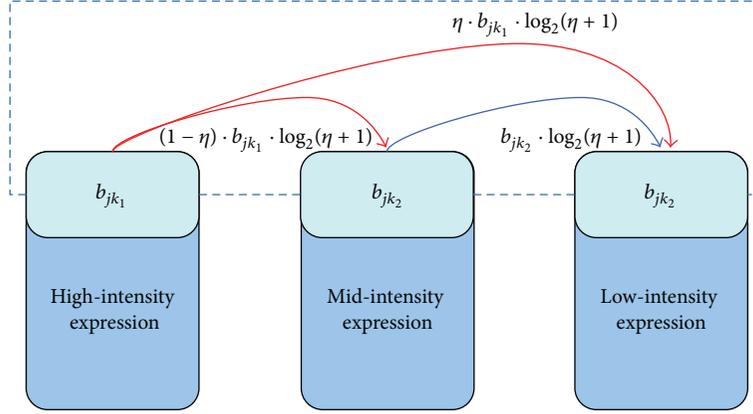


FIGURE 3: Structure of HMM.


 FIGURE 4: The change of  $b_{jk}$  by the personality suppression of expression parameter  $\eta$ .

3.2. *Effect of Personality on Observation State Transition Probability Matrix.* It will get an emotional state called hidden state  $S$  which is recorded as  $t + 1$ , after emotional state is updated by Markov chain. Robots make certain emotional behavior according to that an emotional state and pass it to emotional expression system. To a certain particular emotional status, its expression is different by its inner character. For instance, when it happens to a same happy state, to those open and active personality individuals, the facial expression is much more obvious and vivid compared to those introversion and sensitive personality individuals. So after gaining individual's emotional state, we still need one random process to calculate expression output, which is the general random process in HMM.

The observation state transition probability matrix  $\mathbf{B}$  refers to the probability of 7 kinds of expressions under 7 emotions:

$$\mathbf{B} = (b_{jk})_{7 \times 7} = \begin{pmatrix} b_{11} & b_{12} & \cdots & b_{17} \\ b_{21} & b_{22} & \cdots & b_{27} \\ \vdots & \vdots & \ddots & \vdots \\ b_{71} & b_{72} & \cdots & b_{77} \end{pmatrix}, \quad \sum_{k=1}^7 b_{jk} = 1. \quad (8)$$

Expect for calming, the other 6 expressions are divided into high-intensity, mid-intensity, and low-intensity expression. Thus  $\mathbf{B}$ , is a  $7 \times 19$  matrix:

$$\mathbf{B} = (b_{jk_l})_{7 \times 19} = \begin{pmatrix} b_{11_1} & b_{11_2} & b_{11_3} & b_{12_1} & \cdots & b_{17_1} \\ b_{21_1} & b_{21_2} & b_{21_3} & b_{22_1} & \cdots & b_{27_1} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ b_{71_1} & b_{71_2} & b_{71_3} & b_{72_1} & \cdots & b_{77_1} \end{pmatrix}, \quad (9)$$

$$\sum_{k=1}^6 \sum_{l=1}^3 (b_{jk_l} + b_{j7}) = 1. \quad (10)$$

Here,  $l$  refers to intensity of emotion, and  $l = 1, l = 2, l = 3$  represents separately high-intensity, mid-intensity, and low-intensity expressions  $b_{jk_l} (1 \leq j \leq 7, 1 \leq k \leq 6)$  is the probability that  $k$ th kind of expression with  $l$ th intensity generates under  $j$ th emotion, and  $b_{j7} (1 \leq j \leq 7)$  is the probability of calming under  $j$ th emotion.

The effect that personality has on suppression strategy performs in external expression which refers to the observation state transition probability matrix  $\mathbf{B}$ . In the paper,  $b_{jk}$  is changed by the effect of the personality suppression of expression parameter  $\eta$  as shown in Figure 4.

In Figure 4, the red arrow indicates the sum of transition probability from high-intensity expression to mid-intensity and low-intensity expressions which is  $b_{jk_1} \cdot \log_2(\eta + 1)$ , including the probability  $(1 - \eta) \cdot b_{jk_1} \cdot \log_2(\eta + 1)$  from high- to mid-intensity expression and  $\eta \cdot b_{jk_1} \cdot \log_2(\eta + 1)$  from high- to low-intensity expression. This indicates that with bigger  $\eta$ , the transition probability from high- to mid-intensity expression is smaller, and the transition probability from high- to low-intensity expression is bigger. That is, the stronger the ability of self-control is, the more possible the low-intensity expression happens. On the contrary, with smaller personality suppression of expression parameter  $\eta$ , the sum of transition probability from high-intensity expression to mid-intensity and low-intensity expression is smaller. That is, the weaker the ability of self-control is, the more closed the expression is to the inner experience. The blue arrow indicates the probability from mid- to low-intensity expression, and the meaning is when the personality suppression of expression parameter  $\eta$  is bigger, low-intensity expression performs more possibly.

The mathematic equations can be derived as follows:

$$b'_{jk_l} = \begin{cases} b'_{jk_1} = b_{jk_1} - b_{jk_1} \cdot \log_2(\eta + 1) \\ b'_{jk_2} = b_{jk_2} + (1 - \eta) \cdot b_{jk_1} \cdot \log_2(\eta + 1) \\ \quad - b_{jk_2} \cdot \log_2(\eta + 1) \\ b'_{jk_3} = b_{jk_3} + \eta \cdot b_{jk_1} \cdot \log_2(\eta + 1) + b_{jk_2} \cdot \log_2(\eta + 1). \end{cases} \quad (11)$$

Here,  $1 \leq j \leq 7$ ,  $1 \leq k \leq 6$ , and the probability of calming has not changed,  $b'_{j7} = b_{j7}$ .

It can be proved that

$$\begin{aligned} \sum_{k=1}^7 b'_{jk} &= \sum_{k=1}^6 \sum_{l=1}^3 b'_{jk_l} + b'_{j7} = \sum_{k=1}^6 b'_{jk_1} + \sum_{k=1}^6 b'_{jk_2} + \sum_{k=1}^6 b'_{jk_3} + b'_{j7} \\ &= \sum_{k=1}^6 (b_{jk_1} - b_{jk_1} \cdot \log_2(\eta + 1)) \\ &\quad + \sum_{k=1}^6 [b_{jk_2} + (1 - \eta) \cdot b_{jk_1} \cdot \log_2(\eta + 1) \\ &\quad \quad - b_{jk_2} \cdot \log_2(\eta + 1)] \\ &\quad + \sum_{k=1}^6 [b_{jk_3} + \eta \cdot b_{jk_1} \cdot \log_2(\eta + 1) \\ &\quad \quad + b_{jk_2} \cdot \log_2(\eta + 1)] + b_{j7} \\ &= \sum_{k=1}^6 [b_{jk_1} - b_{jk_1} \cdot \log_2(\eta + 1) + b_{jk_2} \\ &\quad + (1 - \eta) \cdot b_{jk_1} \cdot \log_2(\eta + 1) - b_{jk_2} \cdot \log_2(\eta + 1) \\ &\quad + b_{jk_3} + \eta \cdot b_{jk_1} \cdot \log_2(\eta + 1) \\ &\quad + b_{jk_2} \cdot \log_2(\eta + 1)] + b_{j7} \\ &= \sum_{k=1}^6 [b_{jk_1} + b_{jk_2} + b_{jk_3}] + b_{j7} \\ &= \sum_{k=1}^6 \sum_{l=1}^3 (b_{jk_l} + b_{j7}) = 1. \end{aligned} \quad (12)$$

Corrected emotional expression matrix satisfies the constraint:  $\sum_{k=1}^M b_{jk} = 1$ . Although this model is proposed in the case of the expression intensity divided into three categories, it also applies to other classification of expression intensity.

## 4. Experiment

**4.1. Simulation Results.** Ekman's six basic emotions include fear, anger, sadness, disgust, happiness, and surprise. To describe the emotion transfer and expression of the robot, the calming emotion is added in this model. Thus, emotion state set  $S = \{S_1, S_2, S_3, S_4, S_5, S_6, S_7\}$ , here  $S_1 \sim S_6$  represent

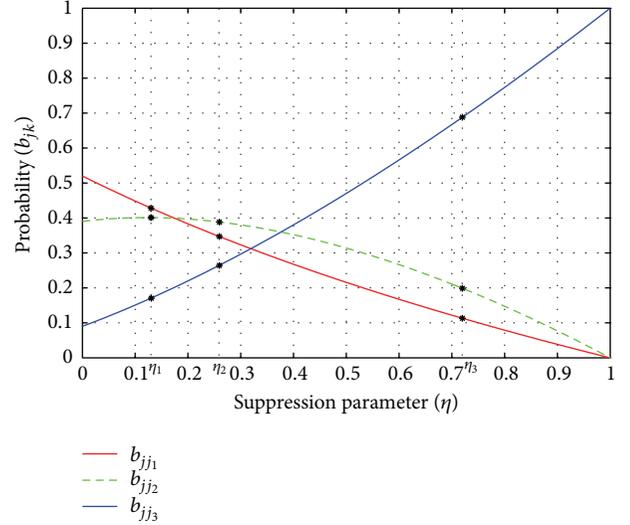


FIGURE 5: The change of probability  $b_{jji}$  under initial probability distribution of sadness  $[b_{33_1}, b_{33_2}, b_{33_3}] = [0.52, 0.39, 0.09]$ :  $b_{jji}$  decreases, and  $b_{jjs}$  increases gradually, and  $b_{jj2}$  decreases after increasing a little. To the individual with the personality suppression of expression parameter  $\eta_1 = 0.13$ , his expression intensity is high, and to the individual with  $\eta_2 = 0.26$ , his expression is mid intensity, and to the third one with  $\eta_3 = 0.72$ , the expression is low intensity.

fear, anger, sadness, disgust, happiness, and surprise, and  $S_7$  is calming. First, we assume that the probability distribution of the hidden state  $S = [q_1, q_2, q_3, q_4, q_5, q_6, q_7]$  is known, and the personality vectors  $\mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3$  of three people are obtained by NEO-PI-R scales. Then, the personality suppression of expression parameters  $\eta_1, \eta_2, \eta_3$  is calculated by formula (6). The observation state transition probability matrix  $\mathbf{B}$  is corrected by  $\eta_1, \eta_2, \eta_3$ , respectively, and the correction results of the expression occurrence probability in different intensity of different individuals are simulated. Assuming that to a certain hidden state  $S_j$ , only the expression corresponding to the emotion is generated,  $j = k$ . Here, Figures 5, 6, and 7 are the simulation results of sadness expression ( $k = 3$ ) correction probability.

Figures 5–7 show that with the increase of the personality suppression of expression parameter, that is, the increase in self-control ability, the probability of generating high-intensity expression decreases, and probability of generating low-intensity expression increases. When the emotion is sadness ( $j = 3$ ) and initial probability distribution of sadness is different, different intensity expressions appear to the same person; under the same distribution of initial probability, the expression intensity is also different to individuals with different personalities.

**4.2. Experience on the Robot Platform.** Expression is the most direct manifestation of human emotional states. In the model of the robot's emotion regulation, HMM is used to realize the suppression strategy in Gross's emotion regulation model. With the same initial probability distribution, robots with different personalities have different intensity expressions;

TABLE 2: Robot's expressions in different intensities.

	Low-intensity	Mid-intensity	High-intensity
Happiness			
Sadness			
Fear			
Anger			
Surprise			
Disgust			
Calming			

TABLE 3: The expression results of robots.

Hidden state	Robot without personality	Robot with personality
Happiness	High-intensity expression	Mid-intensity expression
Fear	High-intensity expression	Low-intensity expression
Disgust	Mid-intensity expression	Low-intensity expression
Sadness	High-intensity expression	Mid-intensity expression
Surprise	Mid-intensity expression	Mid-intensity expression
Anger	High-intensity expression	Low-intensity expression

robots with same personality may have different expression outputs because of different initial probability distributions. In Table 2, different intensity expressions of robot are shown.

In order to verify the actual moderating effects of the model, we apply the personality suppression model on the robot platform and obtain different outputs of expressions using suppression strategy. Three experiences are done in this paper to prove the validity of the model.

*4.2.1. Experience 1.* In this experience, the observation state transition probability matrix  $\mathbf{B}$  is fixed. The expression results of robots with and without personality are recorded in Table 3. The results show that comparing with the robot without personality, the expression of the robot which has

personalities is suppressed and that is more in line with the human communication.

*4.2.2. Experience 2.* The observation state transition probability matrix  $\mathbf{B}$  and the hidden state  $S$  are fixed in the experience. Five groups of personality vectors are inputted to the robot emotion regulation system. Each group contains 100 different personality vectors, and the five groups have high value in the factor of openness, conscientiousness, extraversion, agreeableness, and neuroticism separately. The statistic expression results of robots with different personalities are recorded as shown in Table 4. It can be inferred from the results that the extraversion trait in personality influences the expression intensity more than other traits, and

TABLE 4: Different expression results of robots with different personalities.

Personality trait	High-intensity expression	Mid-intensity expression	Low-intensity expression
Openness	19%	44%	37%
Conscientiousness	53%	35%	12%
Extraversion	6%	19%	75%
Agreeableness	39%	33%	28%
Neuroticism	17%	39%	44%

TABLE 5: Outputs of robots with different observation state transition probability matrices.

Maximum in $[b_{jj_1}, b_{jj_2}, b_{jj_3}]$	Personality vector $\mathbf{P}_1$	Personality vector $\mathbf{P}_2$	Personality vector $\mathbf{P}_3$
$b_{jj_1}$	High-intensity expression	Mid-intensity expression	Low-intensity expression
$b_{jj_2}$	Mid-intensity expression	Low-intensity expression	Low-intensity expression
$b_{jj_3}$	Low-intensity expression	Low-intensity expression	Low-intensity expression

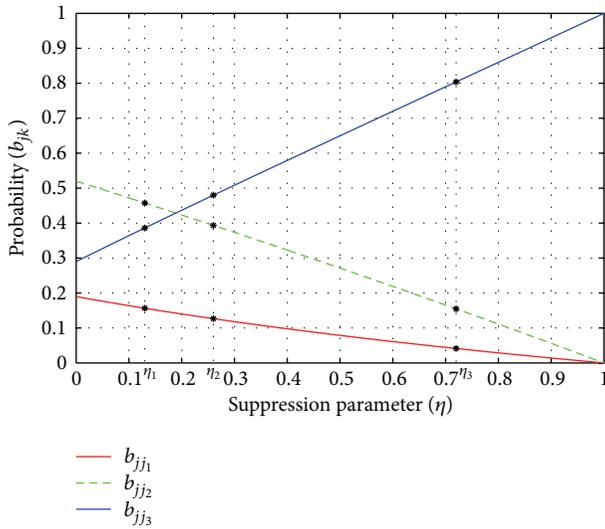


FIGURE 6: The change of probability  $b_{jj_i}$  under initial probability distribution of sadness  $[b_{33_1}, b_{33_2}, b_{33_3}] = [0.19, 0.52, 0.29]$ :  $b_{jj_1}$  decreases more slowly than  $b_{jj_2}$ , and  $b_{jj_3}$  increases gradually. To the individual with the personality suppression of expression parameter  $\eta_1 = 0.13$ , his expression intensity is mid, and to the individual with  $\eta_2 = 0.26$  and the third one with  $\eta_3 = 0.72$ , the expression is low intensity.

the conclusion conforms to the point proposed by Gross and John [9].

4.2.3. *Experience 3.* In experience 3, the hidden state  $S$  is fixed, and there are three different personality vectors. The robots' outputs with different observation state transition probability matrices are recorded in Table 5. The results show that to the same robot with personality, the expression is different because of the different observation state transition probability matrices.

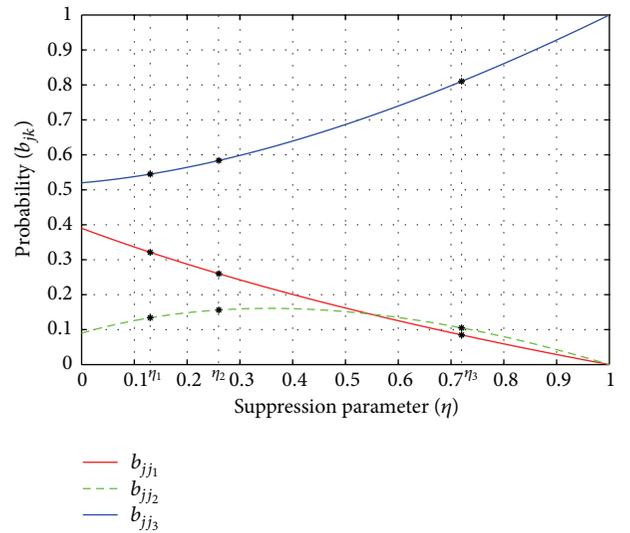


FIGURE 7: The change of probability  $b_{jj_i}$  under initial probability distribution of sadness  $[b_{33_1}, b_{33_2}, b_{33_3}] = [0.39, 0.09, 0.52]$ :  $b_{jj_1}$  decreases, and  $b_{jj_3}$  increases gradually, and  $b_{jj_2}$  decreases after increasing a little. To the individual with the personality suppression of expression parameter  $\eta_1 = 0.13$ , with  $\eta_2 = 0.26$  and with  $\eta_3 = 0.72$ , the expression is low intensity.

Figure 8 shows the typical emotional expressions of the robot. From the three experiences, it can be indicated that the emotional expression suppression model with personality is consistent with the simulation results basically. That means the model proposed in this paper is validated, and human-computer interaction can be improved by using the model.

## 5. Conclusion

Firstly, this paper analyses the importance of the robot's emotional regulation strateg, and applies the Gross' emotion

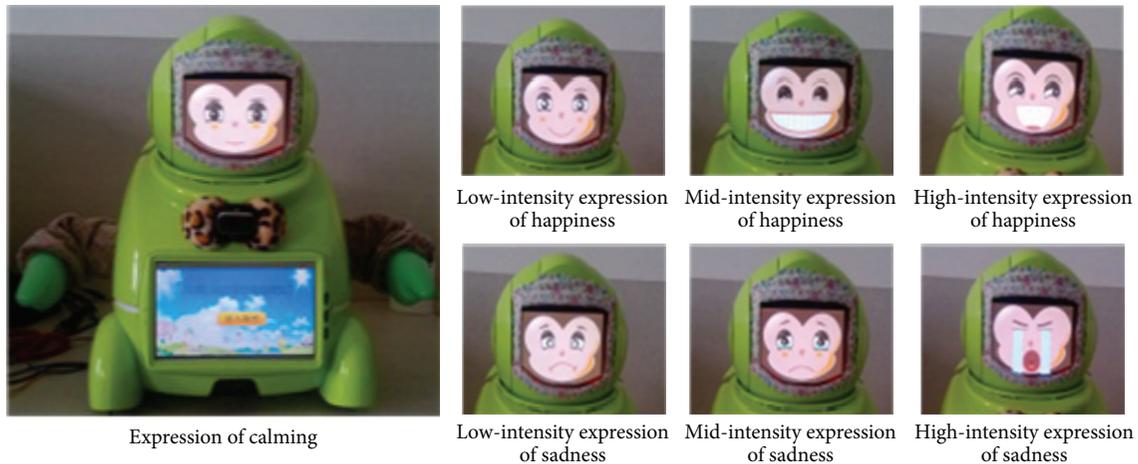


FIGURE 8: Typical emotional expressions of the robot.

regulation model as the psychological basis of robot emotional model. Using HMM algorithm, we deal with the psychological problems mathematically and establish the robot's emotional regulation model. Aiming at the expressional suppression strategy in the process of emotional regulation, we bring in robot personality as an impact factor and then adjust the robot expression matrix. The impact factor does not influence the value of emotional intensity but the occurrence probabilities of different emotional intensities; thus, there is no need to calculate the intensities of different emotions. Using the proposed emotion regulation model in actual robot platform, it can be seen that the robot is more natural and human-like in the human-computer interaction. The statistical analysis of the model analysis and the actual interaction results show that after the introduction of robot personality suppression of expression parameter, anthropomorphic robot is improved, and the effect of human-computer interaction fits the diversity of human emotions.

Because of the sophistication and versatility of human emotion regulation process, many factors affect the regulation. In this paper, we only consider the robot personality, the obvious and influential factor, and cannot fully simulate the human emotion interaction. How to take into account various factors to optimize the model and how to apply the model in the continuous emotional intensity space is one of the main future researches.

## Acknowledgments

This work is supported by National Natural Science Foundation of China (no. 61170115; no. 61170117; no. 61105120), and the 2012 Ladder Plan Project of Beijing Key Laboratory of Knowledge Engineering for Materials Science (no. Z121101002812005).

## References

- [1] J. J. Gross, "Emotion regulation: affective, cognitive, and social consequences," *Psychophysiology*, vol. 39, no. 3, pp. 281–291, 2002.
- [2] R. A. Thompson, "Emotional regulation and emotional development," *Educational Psychology Review*, vol. 3, no. 4, pp. 269–307, 1991.
- [3] J. J. Gross, "Emotion regulation: past, present, future," *Cognition and Emotion*, vol. 13, no. 5, pp. 551–573, 1999.
- [4] M. A. Southam-Gerow and P. C. Kendall, "Emotion regulation and understanding: implications for child psychopathology and therapy," *Clinical Psychology Review*, vol. 22, no. 2, pp. 189–222, 2002.
- [5] P. Xiaolan, X. Lun, L. Xin, and W. Zhiliang, "An algorithm model for gross cognitive reappraisal strategy," *Advances in Intelligent Systems and Computing*, vol. 1, pp. 435–446, 2013.
- [6] T. Bosse, M. Pontier, and J. Treur, "A computational model based on Gross' emotion regulation theory," *Cognitive Systems Research*, vol. 11, no. 3, pp. 211–230, 2010.
- [7] A. Soleimani and Z. Kobti, "An adaptive computational model of emotion regulation strategies based on Gross theory," in *Proceedings of the 5th Conference on Computer Science & Software Engineering*, pp. 9–17, 2012.
- [8] E. Leon, I. Montalban, S. Schlatter, and I. Dorransoro, "Computer-mediated emotional regulation: detection of emotional changes using non-parametric cumulative sum," in *Proceedings of the 32nd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC '10)*, pp. 1109–1112, September 2010.
- [9] J. J. Gross and O. P. John, "Individual differences in two emotion regulation processes: implications for affect, relationships, and well-being," *Journal of Personality and Social Psychology*, vol. 85, no. 2, pp. 348–362, 2003.
- [10] R. R. McCrae and O. P. John Jr., "An introduction to the five-factor model and its applications," *Journal of Personality*, vol. 60, no. 2, pp. 175–215, 1992.
- [11] L. Xin, X. Lun, W. Zhiliang, and F. Dongmei, "Robot emotion and performance regulation based on HMM," *International Journal of Advanced Robotic Systems*, vol. 10, pp. 1–6, 2013.
- [12] L. Pena, J.-M. Pena, and S. Ossowski, "Representing emotion and mood states for virtual agents," in *Proceedings of the 9th German Conference on Multi-Agent System Technologies (MATES '11)*, pp. 181–188, 2011.
- [13] J. Woo Park, W. Hyun Kim, W. Hyong Lee, J. Chang Kin, and M. Jin Chung, "How to completely use the PAD space for socially

- interactive robots,” in *Proceedings of the IEEE International Conference on Robotics and Biomimetics*, pp. 3005–3010, 2011.
- [14] P. Ekman, “Universals and cultural differences in facial expressions of emotion,” in *Proceedings of the Nebraska Symposium on Motivation*, J. Cole, Ed., vol. 19, pp. 207–282, University of Nebraska Press, Lincoln, Nebraska, 1971.
- [15] L. Xie, Z. Wang, and J. Xu, “Mechanical design and affective interaction of bionic robot head,” *Advanced Science Letters*, vol. 4, no. 4-5, pp. 1337–1341, 2011.

## Research Article

# Multiagent Reinforcement Learning with Regret Matching for Robot Soccer

Qiang Liu,<sup>1,2</sup> Jiachen Ma,<sup>1,2</sup> and Wei Xie<sup>1,2</sup>

<sup>1</sup> School of Astronautics, Harbin Institute of Technology, Harbin 150001, China

<sup>2</sup> School of Information and Electrical Engineering, Harbin Institute of Technology (Weihai), Weihai 264209, China

Correspondence should be addressed to Qiang Liu; [hit.liuqiang@163.com](mailto:hit.liuqiang@163.com)

Received 4 April 2013; Revised 19 July 2013; Accepted 20 July 2013

Academic Editor: Yudong Zhang

Copyright © 2013 Qiang Liu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper proposes a novel multiagent reinforcement learning (MARL) algorithm Nash-Q learning with regret matching, in which regret matching is used to speed up the well-known MARL algorithm Nash-Q learning. It is critical that choosing a suitable strategy for action selection to harmonize the relation between exploration and exploitation to enhance the ability of online learning for Nash-Q learning. In Markov Game the joint action of agents adopting regret matching algorithm can converge to a group of points of no-regret that can be viewed as coarse correlated equilibrium which includes Nash equilibrium in essence. It can be inferred that regret matching can guide exploration of the state-action space so that the rate of convergence of Nash-Q learning algorithm can be increased. Simulation results on robot soccer validate that compared to original Nash-Q learning algorithm, the use of regret matching during the learning phase of Nash-Q learning has excellent ability of online learning and results in significant performance in terms of scores, average reward and policy convergence.

## 1. Introduction

Multi-robot system (MRS) has received more and more attention because of its broad application prospect, which has several research platforms including formation [1], foraging [2], prey-pursuing [3, 4], and robot soccer [5–7]. Robot soccer is associated with robot architecture, cooperation, decision making, planning, modeling, learning, vision tracking algorithm, sensing, and communication, which owns all the key features of MRS. And the robot soccer system is discussed as a test benchmark in this paper [8].

Though reinforcement learning (RL), for example, Q-learning [9–11] can be directly applied in MRS for decision-making, it violates the static environment assumption of Markov Decision Process (MDP) [12]. For MRS action selection of the learning robot is unavoidably affected by actions of other agents, so multiagent reinforcement learning (MARL) involving joint state and joint action is more suitable and promising method for MRS [13–16].

MARL based on Stochastic Game (SG) that can be also called Markov game (MG) has a solid theoretical foundation for MRS, which has developed several branches such

as MiniMax-Q learning [17], Nash-Q learning [18], FF-Q learning [19], and CE-Q learning algorithms [20]. Agents adopting the above algorithms can also be called equilibrium learners [17, 20, 21], which is one method of handling the loss of stationarity of MDP. These algorithms learn joint action values which are stationary and in certain circumstances guarantee that these values can converge to Nash equilibrium (NE) values [22] or correlated equilibrium (CE) values. Using these values, the agents' policy corresponds to the agent's component of some Nash or correlated equilibrium [23]. So based on the fundamental solution concept of NE for MG, Nash-Q learning algorithm that finds NE at each state in order to obtain NE policies for Q value updating is an effective and typical MARL method.

For single agent learning scenario, Q-learning is guaranteed to converge to the optimal action independent of the action selection strategy. However, in a multiagent setting, the action selection policy becomes crucial for convergence to any joint action. A big challenge in defining a suitable strategy for the selection of actions is to strike a balance between exploring the usefulness of actions that have been attempted only a few times and exploiting those in which the agents'

confidence in obtaining a high reward is relatively strong. This is known as the exploration and exploitation problem [24].

Regret matching can better harmonize the relation between exploration and exploitation. Regret has been studied both in game theory [25] and computer science [26, 27]. Regret measures how much worse an algorithm performs compared with the best static strategy whose goal is to guarantee at least zero average regret [23]. Regret matching [25] belonging to no-regret algorithms guarantees that the joint action will asymptotically converge to a set of points of no-regret that can be referred to as coarse correlated equilibrium in MG [28]. Because Nash equilibrium is in fact coarse correlated equilibrium [28], it can be inferred that regret matching that leads joint action to points of coarse correlated equilibrium can effectively improve the convergence rate of original Nash-Q learning algorithm.

This paper is organized as follows. Section 2 reviews multiagent reinforcement learning and Nash-Q learning algorithm. Section 3 briefly describes regret matching algorithm and then shows how to incorporate regret matching technique into original Nash-Q learning algorithm. Section 4 describes the structure of reinforcement learning of soccer robot. Section 5 presents simulation demonstration of our algorithm in robot soccer. Section 6 draws a conclusion and summarizes some important points about this paper.

## 2. Multiagent Reinforcement Learning and Nash-Q Learning

**2.1. Markov Game.** Markov game (MG) can be viewed as an extension of MDP to multiagent environments [29, 30], where all agents select their actions simultaneously. The reward that each agent gets depends on their joint action of all agents and the current state as well as the state transitions according to the Markov property [31]. MG is the theory foundation of MARL and Figure 1 shows the architecture. A reinforcement framework of MG can be defined by the following.

An  $n$ -agent MG  $\Gamma$  is a tuple  $\langle n, S, A_1, \dots, A_n, T, r_1, \dots, r_n \rangle$ , where  $n$  represents the number of agents,  $S$  is the state space,  $A_i$  is the action space of agent  $i$  ( $i = 1, \dots, n$ ),  $T: S \times A_1 \times \dots \times A_n \times S \rightarrow \Delta(S)$  is the transition function which depends on the actions of all agents and  $\Delta(S)$  is the set of probability distributions over state space  $S$ , and  $r_i: S \times A_1 \times \dots \times A_n \times S \rightarrow R$  is the reward function for agent  $i$  which also depends on the actions of all agents. Given state  $s$ , each agent independently chooses corresponding action  $a_1, \dots, a_n$  and then receives rewards  $r_i(s, a_1, \dots, a_n)$ ,  $i = 1, \dots, n$ . The next state  $s'$  arrives after joint action  $(a_1, \dots, a_n)$  is taken at state  $s$  based on fixed transition probabilities and the following equation is satisfied:

$$\sum_{s' \in S} P(s', S, a_1, \dots, a_n) = 1. \quad (1)$$

In a discounted MG, the objective of each agent is to maximize the discounted sum of rewards with discount factor

$\beta \in [0, 1)$ . Denote  $\pi_i$  as the strategy of agent  $i$ . For a given initial state  $s$ , agent  $i$  tries to maximize

$$V_i(s, \pi_1, \pi_2, \dots, \pi_n) = \sum_{t=0}^{\infty} \beta^t E(r_1^t | \pi_1, \pi_2, \dots, \pi_n, s_0 = s). \quad (2)$$

**2.2. Comparing among Existing Algorithms.** The traditional Q-learning algorithm [9] for computing an optimal policy in an MDP with unknown reward and transition functions is as follows:

$$\begin{aligned} Q(s, a) &\leftarrow (1 - \alpha) Q(s, a) + \alpha [r(s, a) + \beta V(s')], \\ V(s) &\leftarrow \max_{a \in A} Q(s, a). \end{aligned} \quad (3)$$

The simplest way to extend this to the multiagent MG setting is just to add a subscript to the formulation above and the definition of the  $Q$  values assumes that they depend on the joint action of all agents. Meanwhile  $V$  should be updated with computation outcome of the  $Q$  values corresponding to respective algorithm.

The Minimax-Q learning algorithm as the first MARL extends the traditional Q-learning to the domain of two-player zero-sum multiagent MG environment. In Minimax-Q learning,  $V$  is updated with the minimax of the  $Q$  values:

$$V_1(s) \leftarrow \max_{p_1 \in \Pi(A_1)} \min_{a_2 \in A_2} \sum_{a_1 \in A_1} p_1(a_1) Q_1(s, a_1, a_2). \quad (4)$$

The policy used in the Minimax-Q learning algorithm can guarantee that it receives the largest value possible in the absence of knowledge of the opponent's policy.

Hu and Wellman [21] extended the Minimax-Q algorithm to  $n$ -player general-sum MG. The extension requires that each agent maintains  $Q$  values for all of the agents. And the linear programming solution used to find the equilibrium of zero-sum games is replaced by the quadratic programming solution for finding an equilibrium in  $n$ -player general-sum games. Nash-Q updates the  $V$  values based on some NE in the game defined by the  $Q$ -values:

$$V_i(s) \leftarrow \text{Nash}_i(Q_1(s, a), \dots, Q_n(s, a)), \quad (5)$$

where  $Q_i(s, a)$  denotes the payoff matrix to player  $i$  and  $\text{Nash}_i$  denotes the Nash payoff to that player.

Since Nash-Q is limited to zero-sum and common-payoff games in essence, Littman reinterpreted it as the Friend-or-Foe-Q (FF-Q) learning framework [19]. Although FF-Q can be applied in multiple players scenario, for simplicity we show how the  $V$  are updated in a two-player game:

$$\begin{aligned} \text{Friend: } V_1(s) &\leftarrow \max_{a_1 \in A_1, a_2 \in A_2} Q_1(s, a_1, a_2), \\ \text{Foe: } V_1(s) &\leftarrow \max_{p_1 \in \Pi(A_1)} \min_{a_2 \in A_2} \sum_{a_1 \in A_1} p_1(a_1) Q_1(s, a_1, a_2). \end{aligned} \quad (6)$$

Thus Friend-Q updates  $V$  similarly to regular Q-learning, and Foe-Q updates as does minimax-Q.

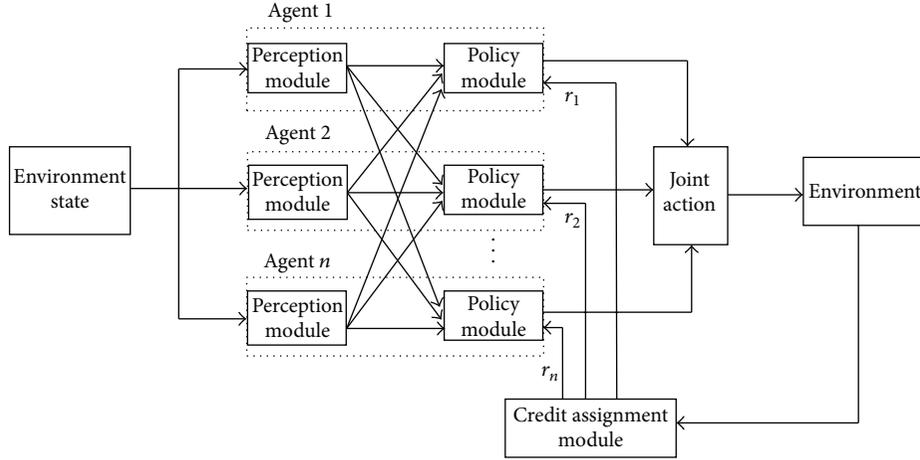


FIGURE 1: Architecture of MARL based on Markov game.

The above algorithms in this section all depend on some methods of computing the NE for the matrix game defined by  $Q$  values of all players in each state. The value for each player of a mutually agreed-on equilibrium is the value function used in the  $Q$  update process. Instead of computing Nash equilibria of  $Q$  stage games, the agent can compute other solution concepts. One option is computing the CE. This is the technique used by Greenwald and Hall in the unambiguously named Correlated-Q (CE-Q) algorithm [20]. A CE is more general than an NE, since it allows dependencies among the agents' probability distributions, while maintaining the property that agents are optimizing. Compared to NE, CE can be computed easily via linear programming. CE-Q learning is similar to Nash-Q but instead uses the value of a correlated equilibrium to update  $V$ :

$$V_i(s) \leftarrow \text{CE}_i(Q_1(s, a), \dots, Q_n(s, a)). \quad (7)$$

Like Nash-Q, it requires agents to select a unique equilibrium; an issue that the authors address explicitly by suggesting several possible selection mechanisms.

**2.3. Nash-Q Learning.** The following is based on [18]. Extending  $Q$ -learning to the multiagent learning domain with NE concept, Nash-Q equilibrium value is defined as the expected sum of discounted rewards when all agents follow specified Nash equilibrium strategies from the next period on. The literature usually uses the terms policy and strategy interchangeably. A Nash equilibrium is a joint strategy where each agent's strategy is a best response to the others' strategies.

In MG  $\Gamma$ , a Nash equilibrium point is a tuple of  $n$  strategies  $(\pi_1^*, \dots, \pi_n^*)$  such that for all  $s \in S$  and  $i = 1, \dots, n$ ,

$$\begin{aligned} v_i(s, (\pi_1^*, \dots, \pi_n^*)) \\ \geq v_i(s, \pi_1^*, \dots, \pi_{i-1}^*, \pi_i, \pi_{i+1}^*, \dots, \pi_n^*) \quad \forall \pi_i \in \Pi_i, \end{aligned} \quad (8)$$

where  $\Pi_i$  is the set of strategies available to agent  $i$ .  $Q_i^*$  is defined as a Nash-Q function for agent  $i$  and  $Q_i^*(s, a_1, \dots, a_n)$  is called Nash-Q equilibrium value. Nash-Q function of agent

$i$  is defined over  $(s, a_1, \dots, a_n)$ , as the sum of agent  $i$ 's current reward plus its future rewards when all agents follow the joint NE strategy. That is,

$$\begin{aligned} Q_i^*(s, a_1, \dots, a_n) \\ = r_i^*(s, a_1, \dots, a_n) \\ + \beta \sum_{s' \in S} P(s' | s, a_1, \dots, a_n) v^i(s', \pi_1^*, \dots, \pi_n^*), \end{aligned} \quad (9)$$

where  $(\pi_1^*, \dots, \pi_n^*)$  is the joint Nash equilibrium strategy,  $r_i(s, a_1, \dots, a_n)$  is agent  $i$ 's one stage reward in state  $s$  and under joint action  $(a_1, \dots, a_n)$ , and  $V_i(s', \pi_1^*, \dots, \pi_n^*)$  is agent  $i$ 's total discounted reward over infinite periods starting from state  $s'$  given that agents follow the equilibrium strategies.

In the case of multiple equilibria, different NE strategy profiles may select different Nash-Q functions. In this paper, the learning agent picks the NE that yields the highest expected payoff to them as a whole. The learning agent indexed by  $i$  learns about its  $Q$  values by forming an arbitrary guess at time 0. One simple guess would be letting  $Q_i^0(s, a_1, \dots, a_n) = 0$  for all  $s \in S, a_1 \in A_1, \dots, a_n \in A_n$ . At each time  $t$ , agent  $i$  observes the current state and then takes its action. After actions were taken, agent  $i$  observes its own reward, actions taken by all other agents, others' rewards, and the new state  $s'$ . It then calculates a Nash equilibrium  $\pi_1(s') \cdots \pi_n(s')$  and updates its  $Q$  values according to

$$\begin{aligned} Q_i^{t+1}(s, a_1, \dots, a_n) = (1 - \alpha_i) Q_i^t(s, a_1, \dots, a_n) \\ + \alpha_i [r_i^t + \beta \text{Nash} Q_i^t(s')], \end{aligned} \quad (10)$$

where  $\text{Nash} Q_i^t(s') = \pi_1(s') \cdots \pi_n(s') \cdot Q_i^t(s')$ .

$\text{Nash} Q_i^t(s')$  is agent  $i$ 's payoff in state  $s'$  for the selected equilibrium. Note that  $\pi_1(s') \cdots \pi_n(s') \cdot Q_i^t(s')$  is a scalar. The learning algorithm is as follows:

Initialize:

Let  $t = 0$ , get the initial state  $s_0$ ;

Let the learning agent be indexed by  $i$ ;

For all  $s \in S$  and  $a_j \in A_j$ ,  $j = 1, \dots, n$ , let  $Q_j^t(s, a_1, \dots, a_n) = 0$ .

Loop

Choose action  $a_i^t$ ;  
Observe  $r_1^t, \dots, r_n^t$ ;  $a_1^t, \dots, a_n^t$ , and  $s^{t+1} = s'$ ;  
Update  $Q_j^t$  for  $j = 1, \dots, n$ .

$$Q_j^{t+1}(s, a_1, \dots, a_n) = (1 - \alpha_t) Q_j^t(s, a_1, \dots, a_n) + \alpha_t [r_j^t + \beta \text{Nash} Q_j^t(s')], \quad (11)$$

where  $\alpha_t \in (0, 1)$  is the learning rate, and  $\text{Nash} Q_j^t(s')$  is defined in (10).

Let  $t := t + 1$ .

For obtaining the NE  $\pi_1(s') \dots \pi_n(s')$ , agent  $i$  need to know  $Q_1^t(s'), \dots, Q_n^t(s')$ . Agent  $i$  should have conjectures about those  $Q$ -functions at the beginning of play. As the game proceeds, agent  $i$  observes other agents' immediate rewards and previous actions. That information can then be used to update agent  $i$ 's conjectures on other agents'  $Q$ -functions. Agent  $i$  updates its beliefs about agent  $j$ 's  $Q$ -function, according to the same updating rule (10) it applies to its own:

$$Q_j^{t+1}(s, a_1, \dots, a_n) = (1 - \alpha_t) Q_j^t(s, a_1, \dots, a_n) + \alpha_t [r_j^t + \beta \text{Nash} Q_j^t(s')]. \quad (12)$$

Note that  $\alpha_t = 0$  for  $(s, a_1, \dots, a_n) \neq (s_t, a_1, \dots, a_n)$ . Therefore (12) does not update all the entries in the  $Q$ -functions. It updates only the entry corresponding to the current state and actions chosen by the agents. This type of updating is called asynchronous updating [18].

### 3. Regret Matching Algorithm for Action Selection

By observing human ways of handling problems, we can conclude that a human often reflects how regretful it is for the decision that he had made. Through reflecting on past action and feeling regretful, a human can learn more experience, find improved action under complicated environment, and enhance the learning efficiency. Regret enables him to obtain better policy and to make progress quickly. In case that people of community all adopt such idea, then the joint action will bring each one good reward.

Based on the above notion, no regret learning algorithms are proposed and have been widely studied and applied in multiagent learning. No regret learning algorithms consist of a lot of algorithms which guarantee that the joint action will converge asymptotically to a set of points of no-regret that can also be called coarse correlated equilibrium [32]. A no-regret point represents a case for which the average reward which an agent actually obtained is as much as the counterpart that the agent "would have" obtained had that

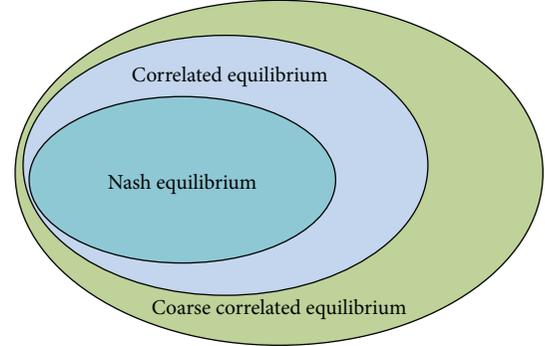


FIGURE 2: Relationship between Nash, correlated, and coarse correlated equilibrium.

agent used a different fixed strategy at all previous time steps [28]. Figure 2 shows that Nash equilibrium belongs to not only correlated equilibrium but also coarse CE. In other words, it is important to note that convergence to a NE point also implies convergence to a coarse correlated equilibrium point (no-regret point).

The prominent feature of regret matching [25] as a branch of no regret learning algorithms is that compared to other learning algorithms, for example, fictitious play [33], it can be easily applied in large scale MRS [28]. The detailed description of regret matching can be found in [25]. And a new algorithm Nash-Q learning with regret matching is proposed to increase the rate of convergence in MG. In the proposed algorithm, regret matching is used to select the action in each state to increase the convergence rate toward Nash equilibrium policy.

According to the above notation, we define the average regret  $R_i^{a_i}(s, t)$  of agent  $i$  at time  $t$  and in state  $s$  as

$$R_i^{a_i}(s, t) = \frac{1}{N} \sum_{m=0}^{N-1} (r_i(s, a_i, a_{-i}(m)) - r_i(s, a(m))), \quad (13)$$

where  $a_{-i}$  denotes the collective  $(a_1, \dots, a_{i-1}, a_{i+1}, \dots, a_n)$ , of agents' action except agent  $i$ ,  $a$  represents the joint action  $(a_1, \dots, a_n)$  of all agents, and  $N$  represents the number of state  $s$  visited.

Equation (13) shows that average regret for  $a_i \in A_i$  of agent  $i$  would represent the average improvement in his reward if it had chosen  $a_i \in A_i$  in all past steps and all other agents' actions had remained unchanged up to time  $t$ . Regret matching based each agent  $i$  computes  $R_i^{a_i}(s, t)$  for every action  $a_i \in A_i$  using the following iterative equation:

$$R_i^{a_i}(s, t) = \frac{t-1}{t} R_i^{a_i}(s, t-1) + \frac{1}{t} (r_i(s, a_i, a_{-i}(t)) - r_i(s, a(t))). \quad (14)$$

Note that at each time step  $t > 0$ , agent  $i$  updates all entries included in his average regret assemble  $R_i(s, t) = [R_i^{a_i}(s, t)]_{a_i \in A_i}$ . In regret matching after agent  $i$  computed its average regret assemble  $R_i(s, t)$ , action  $a_i(s, t)$  is selected

according to the probability distribution  $p_i(t)$ , as shown in the following equation:

$$p_i^{a_i}(t) = \Pr [a_i(s, t) = a_i] = \frac{R_i^{a_i}(s, t)}{\sum_{a'_i \in A_i} [R_i^{a'_i}(s, t)]}, \quad (15)$$

where  $p_i(t)$  is the uniform distribution over  $A_i$ . In other words, an agent using regret matching selects a particular action at any time step with probability proportional to the average regret for not selecting that particular action in the past time steps.

If all agents of one team choose regret matching algorithm for robot soccer, then the joint action will converge asymptotically to a set of points of coarse CE. So it can be inferred that regret matching can effectively improve the convergence rate of original Nash-Q learning, which is validated by the following simulation.

#### 4. Action-Based Soccer Robot

**4.1. Environment States and Joint Action of Robot.** Robot soccer is a very challenging and interesting domain for the application of machine learning algorithms to real world problems. Research groups have applied a lot of different machine learning approaches to many facets of autonomously soccer playing MRS [34].

Behavior-(action-) based approaches are very suited for soccer because they have outstanding performance than deliberative control in uncertain and dynamic environments. Behavior design of the robot (agent) soccer team is based on the following two characteristics. Firstly, points are scored by kicking the ball across the opponent team's goal. Secondly, robots should avoid kicking the ball toward the wrong directions, lest they score against their own team [35].

In this paper, environment states represented in Figure 3 are used to activate the robot. For simplicity each team is composed of three agents (players) as shown in Figure 5. Based on [36], a motor schema-based reactive control system is used for action designing in which each agent is provided three preprogrammed actions (behavior assemblages) that correspond to steps in achieving the task as shown in Table 1. These actions are in turn composed of more primitive behaviors called motor-schemas. Several motor-schemas are described as follows.

**Move\_to\_kickspot:** high gain to draw the robot to a point one-half of a robot radius behind the ball. If the robots bumps the ball from that location, the ball is propelled in the direction of the opponent's goal. **Avoid\_teammates:** gain sufficiently high to keep the robots on the team spread apart. **Move\_to\_half\_point:** high gain to draw the robot to a point halfway between the ball and the defended goal. **Swirl\_ball:** a ball dodging vector with gain sufficiently high to keep the robots from colliding with the ball. **Move\_to\_defended\_goal:** high gain to draw the robot to the defend goal [35].

Shoot ball action is showed in Figure 4. Being analogous to Figure 4, chase ball action is composed of three primitive schemas: **move\_to\_halfway\_point**, **swirl\_ball**, and **avoid\_teammates**. Goal keeping action is composed

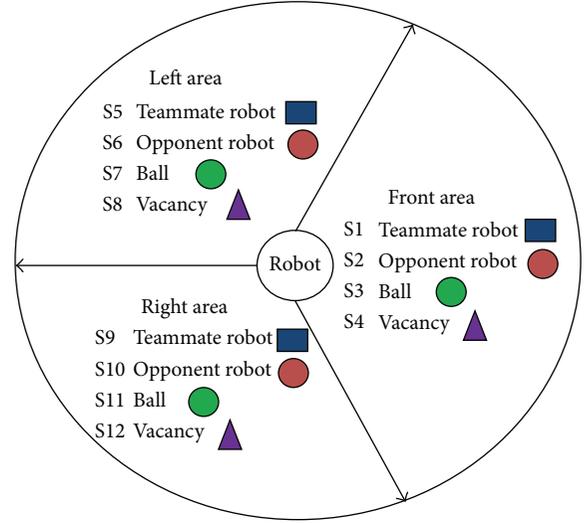


FIGURE 3: Environment states around soccer robot.

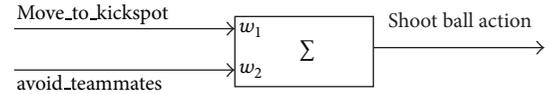


FIGURE 4: Motor schema based on shoot ball action.

of two primitive schemas: **move\_to\_defended\_goal** and **move\_to\_kickspot**.

**4.2. Reward Function for Soccer.** As an instant evaluator the reward function for the action taken at a given state is important for reinforcement learning. Global reinforcement [35] refers to the case where a single reinforcement signal is simultaneously delivered to all robots. A potential problem with global reinforcement is the ambiguous assumption that the closet robot just happened to be near the goal while another soccer robot kicked the ball for a score from a distance. Two important factors should be considered: time and distance, and a modified reward function  $r_i(s, a)$  for agent  $i$  from global reinforcement is as follows:

$$r_i(s, a) = \begin{cases} \eta^{\text{touch}} + \frac{1}{d} & \text{if the team scored at } t-1, \\ -\left(\eta^{\text{touch}} + \frac{1}{d}\right) & \text{if the opponent scored at } t-1, \\ 0 & \text{otherwise,} \end{cases} \quad (16)$$

where  $s$  denotes the soccer robot's state,  $a$  represents the joint action of all agents  $(a_1, \dots, a_n)$ , touch is time in milliseconds since the soccer robot last touched the ball, and  $d$  represents the distance in meters between the ball and robot.  $\eta$  is a parameter value varying between 0.5 and 1 that indicates how quickly a potential reward should decay after the ball is touched, and in this paper  $\eta$  is set to be 0.7.

TABLE 1: Actions of soccer robot [37].

Action	Robot activity
Shoot ball	If robot is close to the ball and goal, this action is used to shoot the ball.
Chase ball	When robot is far away from ball, this action is given to go after the ball.
Goal keeping	Robot playing as a goal keeper gets this action to prevent losing point.



FIGURE 5: Simulated robot soccer.

## 5. Simulation

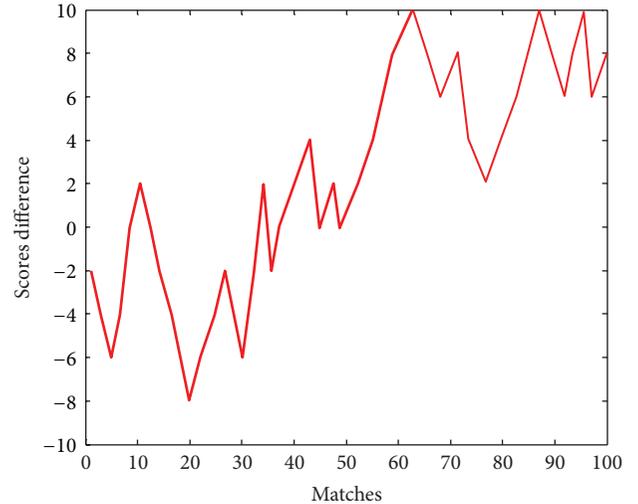
TeamBots as shown in Figure 5 is a Java-based assemble of application programs and java packages for multiagent mobile robotics research, where control system of a robot interacts with a well-defined sensor-actuator interface. The simulation proceeds in discrete steps. The robots process their sensor data in each step and then issue appropriate actuator commands. The simulation models physical interactions including robot, ball and wall collisions, sensors, and motor-driven actuators [24].

Two teams A and B of soccer robots are designed and each team is composed of three agents. Team A adopts Nash-Q learning with regret matching algorithm and Team B is equipped with original Nash-Q learning that learns Nash-Q equilibrium values by random action selection strategy. If a goal is kicked, the ball will be replaced to the center of the field without repositioning the agents and the match goes on. Historical data including scores, average reward, and the average number of policy changes are saved as the match proceeds. The agents preserve Nash-Q values learned between matches. No limited time is imposed on playing that the whole match is not over until a total of 10 points are completed. The simulation is composed of 100 10-point matches. The reward functions that the robots Team A and B adopt are the same as shown in (16).

At the beginning  $Q$  values of all robots were initialized with zero value. An important performance for robot soccer is measured as the scores difference  $D$ :

$$D = S_{\text{teamA}} - S_{\text{teamB}}, \quad (17)$$

where  $S_{\text{teamA}}$  denotes the scores of Team A and  $S_{\text{teamB}}$  is the scores of Team B. A negative value indicates that Team A lost the match, while positive values indicate that Team A won the match. Figure 6 shows the curves of scores difference  $D$  through which we know that robots of Team A found good strategy of joint action resulting in draw or scoring

FIGURE 6: Scores difference  $D$  as the number of matches increases.

over 5 points after the 37th match and outperformed Team B from the 52th match to the end of simulation. It can be summarized that robots of Team A with action exploration strategy of regret matching have accumulated much experience by computing regret value for every action and gradually taking joint action improved after the 37th match. By online learning of regret matching, the joint actions of robots of Team A are gradually close to approach points of coarse correlated equilibrium, which greatly improved the offensive and defensive capabilities of the whole team.

Through Figure 7, it may be concluded that the robots of Team A received positive rewards most of the matches. The average reward per match is increased as matches proceed when the robots obtained more experience of cooperating. It increases from approximately 3.8 to 8.7 in 100 rounds of continuous matches. Because a bigger average reward indicates that the robots have employed good cooperation strategies to kicking more goals, Figure 7 confirms that regret matching as action selection strategy is effective in helping the agents to improve the quality of tactics coordination in carrying out the cooperative attacking. Although for the initial learning phase (the former 37 matches) Team A has worse performance than Team B, as the matches proceed the performance of Team A becomes better and better as we expect before the simulation. For the latter 63 matches, the robots of Team A can quickly adapt themselves to the transition of environment state and coordinate their joint action reducing conflict with their own teammates and obtaining more and more positive rewards.

Learning rate is evaluated by monitoring the policy convergence which is tracked by recording the average number

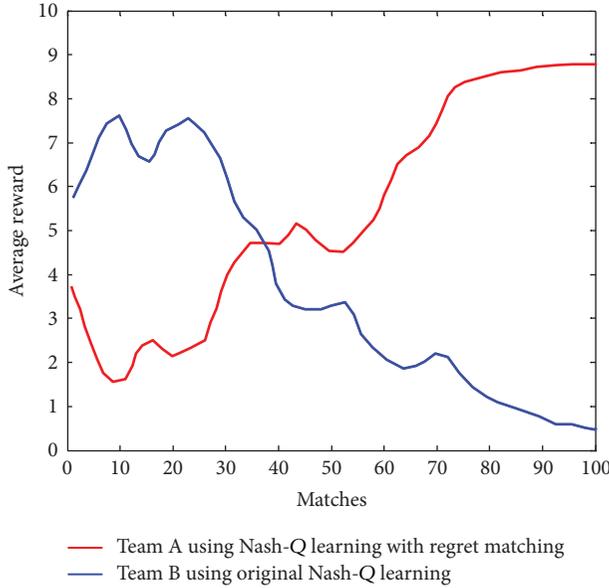


FIGURE 7: The average reward received by the robots in match.

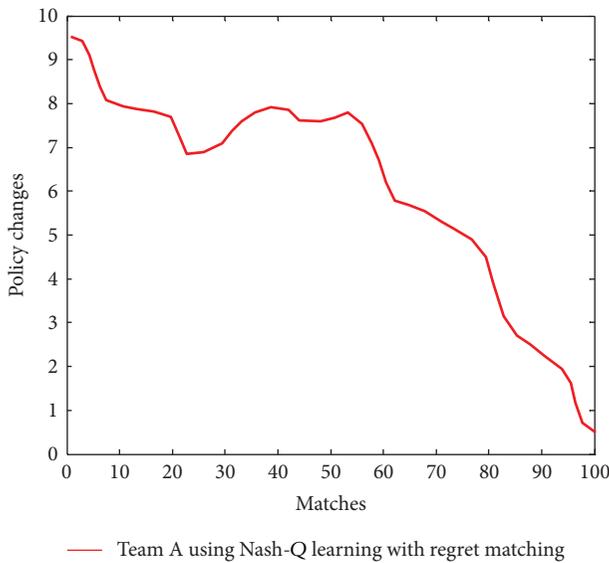


FIGURE 8: The average number of policy changes per match.

of policy changes for all agents of Team A. For example, an agent from Team A may have been following a strategy of goal keeping when opponents appearing in front area, the ball in front area and teammates in left and right, but owing to regret matching it switches to the chase ball action instead. Such alteration is viewed as policy change. The average number of policy changes for Team B is stochastic because of strategy of random action selection. So only the curve of policy changes of Team A is analyzed. The data plotted in Figure 8 shows good convergence for Team A using regret matching algorithm. The average number of policy changes per match dropped to 0.47 after 100 matches.

The number of policy changes for robots of Team A initially is high but decreases gradually in the latter matches.

It can be noted that there is turnpoint at around the 52th match, from which the average number of policy changes of Team A monotonously diminished. An extended simulation shows that the average number of policy changes for Team A reached zero after 150 matches.

From Figures 6 to 8, it is clear that the new Nash-Q with regret matching learning algorithm has higher learning efficiency than the original Nash-Q learning algorithm in robot soccer. Regret matching can better harmonize the tradeoff between exploration and exploit such that the agent can reinforce the evaluation of the actions it already knows to be good but also explore new actions. In particular, the new algorithm Nash-Q learning with regret matching takes an average of 150 matches for completing policy convergence in order to find Nash-Q equilibrium values early.

### 6. Conclusion

This paper presents a new multiagent reinforcement learning approach combining Nash-Q learning with regret matching to increase the convergence rate of original Nash-Q learning algorithm that learns Nash-Q equilibrium values by random action selection in multiagent system. Regret matching which belongs to online learning as a branch of no regret learning algorithms can guarantee that the joint action will asymptotically converge to a set of points of coarse correlated equilibrium including Nash equilibrium points. So we investigate how to make improved action selection in original Nash-Q learning algorithm through regret matching. Robot soccer is adopted as platform to test the proposed approach. Compared to original Nash-Q learning, the results of experiments validate that Nash-Q learning with regret matching algorithm has better performance in terms of scores, average reward, and policy convergence for obtaining the Nash equilibrium policy.

### References

- [1] P. K. C. Wang, "Navigation strategies for multiple autonomous mobile robots moving in formation," *Journal of Robotic Systems*, vol. 8, no. 2, pp. 177–195, 1991.
- [2] M. J. Matarić, "Reinforcement learning in the multi-robot domain," *Autonomous Robots*, vol. 4, no. 1, pp. 73–83, 1997.
- [3] M. Tan, "Multi-agent reinforcement learning: Independent versus cooperative agents," in *Proceedings of the 10th International Conference on Machine Learning*, pp. 330–337, Morgan Kaufmann, 1993.
- [4] T. Fujii, Y. Arai, H. Asama, and I. Endo, "Multilayered reinforcement learning for complicated collision avoidance problems," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 3, pp. 2186–2191, May 1998.
- [5] E. Uchibe, M. Nakamura, and M. Asada, "Co-evolution for cooperative behavior acquisition in a multiple mobile robot environment," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 1, pp. 425–430, October 1998.
- [6] K. H. Park, Y. J. Kim, and J. H. Kim, "Modular Q-learning based multi-agent cooperation for robot soccer," *Robotics and Autonomous Systems*, vol. 35, no. 2, pp. 109–122, 2001.

- [7] Y. Ma, Z. Cao, X. Dong, C. Zhou, and M. Tan, "A multi-robot coordinated hunting strategy with dynamic alliance," in *Proceedings of the Chinese Control and Decision Conference (CCDC '09)*, pp. 2338–2342, chn, June 2009.
- [8] J. H. Kim and P. Vadakkepat, "Multi-agent systems: a survey from the robot-soccer perspective," *Intelligent Automation and Soft Computing*, vol. 6, no. 1, pp. 3–18, 2000.
- [9] C. J. C. H. Watkins and P. Dayan, "Q-learning," *Machine Learning*, vol. 8, no. 3-4, pp. 279–292, 1992.
- [10] M. E. Harmon and S. S. Harmon, *Reinforcement Learning: A Tutorial*, Wright Lab, Wright-Patterson AFB, Ohio, USA, 1997.
- [11] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, MIT Press, Cambridge, UK, 1998.
- [12] Y. Wang, *Cooperative and intelligent control of multi-robot systems using machine learning [thesis]*, The University of British Columbia, 2008.
- [13] Y. Duan, B. X. Cui, and X. H. Xu, "A multi-agent reinforcement learning approach to robot soccer," *Artificial Intelligence Review*, vol. 38, no. 3, pp. 193–211, 2012.
- [14] F. Michaud and M. J. Mataric, "Learning from history for behavior-based mobile robots in non-stationary conditions," *Machine Learning*, vol. 31, no. 1–3, pp. 141–167, 1998.
- [15] M. Asada, E. Uchibe, and K. Hosoda, "Cooperative behavior acquisition for mobile robots in dynamically changing real worlds via vision-based reinforcement learning and development," *Artificial Intelligence*, vol. 110, no. 2, pp. 275–292, 1999.
- [16] M. Wiering, R. Śaustowicz, and J. Schmidhuber, "Reinforcement learning soccer teams with incomplete world models," *Autonomous Robots*, vol. 7, no. 1, pp. 77–88, 1999.
- [17] M. L. Littman, "Markov games as a framework for multi-agent reinforcement learning," in *Proceedings of the 11th International Conference on Machine Learning*, pp. 157–163, 1994.
- [18] J. Hu and M. P. Wellman, "Nash Q-learning for general-sum stochastic games," *Journal of Machine Learning Research*, vol. 4, no. 6, pp. 1039–1069, 2004.
- [19] M. L. Littman, "Friend-or-foe Q-learning in general-sum games," in *Proceedings of the 18th International Conference on Machine Learning (ICML '01)*, pp. 322–328, 2001.
- [20] A. Greenwald and K. Hall, "Correlated-Q learning," in *Proceedings of the 20th International Conference on Machine Learning*, pp. 242–249, August 2003.
- [21] J. Hu and M. P. Wellman, "Multiagent reinforcement learning: theoretical framework and an algorithm," in *Proceedings of the 15th International Conference on Machine Learning*, pp. 242–250, 1998.
- [22] J. Nash, "Non-cooperative games," *The Annals of Mathematics*, vol. 54, no. 2, pp. 286–295, 1951.
- [23] M. Bowling, "Convergence and no-regret in multiagent learning," *Advances in Neural Information Processing Systems*, vol. 17, pp. 209–216, 2005.
- [24] S. Kapetanakis and D. Kudenko, "Reinforcement learning of coordination in cooperative multi-agent systems," in *Proceedings of the National Conference on Artificial Intelligence*, pp. 326–331, AAAI Press, MIT Press, Menlo Park, Calif, USA, 1999.
- [25] S. Hart and A. Mas-Colell, "A simple adaptive procedure leading to correlated equilibrium," *Econometrica*, vol. 68, no. 5, pp. 1127–1150, 2000.
- [26] P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire, "Gambling in a rigged casino: the adversarial multi-armed bandit problem," in *Proceedings of the 36th IEEE Annual Symposium on Foundations of Computer Science*, pp. 322–331, October 1995.
- [27] M. Zinkevich, "Online convex programming and generalized infinitesimal gradient ascent," in *Proceedings of the 20th International Conference on Machine Learning (ICML '03)*, Washington, DC, USA, 2003.
- [28] J. R. Marden, *Learning in Large-Scale Games and Cooperative Control*, University of California, Los Angeles, Los Angeles, Calif, USA, 2007.
- [29] F. Thusijnsman, *Optimality and Equilibrium in Stochastic Games*, Centrum voor Wiskunde en Informatica, 1992.
- [30] X. Wang and T. Sandholm, "Reinforcement learning to play an optimal nash equilibrium in team markov games," *Advances in Neural Information Processing Systems*, vol. 15, pp. 1571–1578, 2002.
- [31] E. Yang and D. Gu, "Multiagent reinforcement learning for multi-robot systems: a survey," University of Essex Technical Report CSM-404, Department of Computer Science, 2004.
- [32] H. P. Young, *Strategic Learning and Its Limits*, Oxford University Press, New York, NY, USA, 2004.
- [33] D. Monderer and L. S. Shapley, "Fictitious play property for games with identical interests," *Journal of Economic Theory*, vol. 68, no. 1, pp. 258–265, 1996.
- [34] M. Riedmiller, T. Gabel, R. Hafner, and S. Lange, "Reinforcement learning for robot soccer," *Autonomous Robots*, vol. 27, no. 1, pp. 55–73, 2009.
- [35] T. Balch, *Behavioral diversity in learning robot teams [thesis]*, Georgia Institute of Technology, 1998.
- [36] R. C. Arkin, "Cooperation without communication: multiagent schemabased robot navigation," *Journal of Robotic Systems*, vol. 9, no. 3, pp. 351–364, 1992.
- [37] T. Y. Tsou, C. H. Liu, and Y. T. Wang, "Term formation control for soccer robot systems," in *Proceeding of the IEEE International Conference on Networking, Sensing and Control*, pp. 1121–1125, March 2004.

## Research Article

# A Wavelet Kernel-Based Primal Twin Support Vector Machine for Economic Development Prediction

Fang Su<sup>1</sup> and HaiYang Shang<sup>2</sup>

<sup>1</sup> College of Economics, Lanzhou University of Technology, 287 Langongping Road, Qilihe District, Lanzhou City 730050, China

<sup>2</sup> Lanzhou University of Finance and Economics, Gansu 730050, China

Correspondence should be addressed to Fang Su; fangsus@sohu.com

Received 17 May 2013; Revised 6 July 2013; Accepted 8 July 2013

Academic Editor: Vishal Bhatnagar

Copyright © 2013 F. Su and H. Shang. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Economic development forecasting allows planners to choose the right strategies for the future. This study is to propose economic development prediction method based on the wavelet kernel-based primal twin support vector machine algorithm. As gross domestic product (GDP) is an important indicator to measure economic development, economic development prediction means GDP prediction in this study. The wavelet kernel-based primal twin support vector machine algorithm can solve two smaller sized quadratic programming problems instead of solving a large one as in the traditional support vector machine algorithm. Economic development data of Anhui province from 1992 to 2009 are used to study the prediction performance of the wavelet kernel-based primal twin support vector machine algorithm. The comparison of mean error of economic development prediction between wavelet kernel-based primal twin support vector machine and traditional support vector machine models trained by the training samples with the 3–5 dimensional input vectors, respectively, is given in this paper. The testing results show that the economic development prediction accuracy of the wavelet kernel-based primal twin support vector machine model is better than that of traditional support vector machine.

## 1. Introduction

As policy makers examine the future economic plans for their regions, economic development forecasting allows planners to choose the right strategies for the future [1, 2]. Gross domestic product (GDP) is an important indicator to measure economic development. Thus, economic development prediction means GDP prediction in this study. Artificial neural networks are the popular prediction algorithms, which have high parallel processing and error tolerance ability [3–5]. Bildirici et al. applied artificial neural networks to economic development prediction, and according to tests of equal forecast accuracy, the results suggest obvious advantages of artificial neural networks compared to regression analysis algorithm [6]. Kim et al. presented early warning system of economic crisis based on artificial neural networks; the experimental results indicated that artificial neural networks can predict economic growth effectively [7]. However, artificial neural networks have the shortcomings of local extremum and overfitting [8–10]. Support vector machine

based on the statistical learning theory has already outperformed most other prediction algorithms [11–14]. This study is to propose economic development prediction method based on the wavelet kernel-based primal twin support vector machine algorithm (WPTSVM). The wavelet kernel-based primal twin support vector machine algorithm can solve two smaller sized quadratic programming problems instead of solving a large one as in the traditional support vector machine algorithm. In the wavelet kernel-based primal twin support vector machine, Morlet wavelet function [15–18] can be used as its kernel function.

Economic development data of Anhui province from 1992 to 2009 are used to study the prediction performance of the wavelet kernel-based primal twin support vector machine algorithm. In this experiment, we employ the training samples with different dimensional input vector to train the wavelet kernel-based primal twin support vector machine algorithm. The comparison of the prediction values between the wavelet kernel-based primal twin support vector machine model and traditional support vectors machine

model trained by the training samples with 3-, 4-, and 5-dimensional input vectors, respectively, is given; and the comparison of the prediction error between the wavelet kernel-based primal twin support vector machine model and traditional support vector machine model trained by the training samples with 3-, 4-, and 5-dimensional input vectors, respectively, is given. And the comparison of mean error of economic development prediction between wavelet kernel-based primal twin support vector machine and traditional support vector machine model trained by the training samples with the 3–5 dimensional input vectors, respectively, is given. It can be seen that the economic development prediction accuracy of the wavelet kernel-based primal twin support vector machine model is better than that of traditional support vector machine.

The organization of this paper has been described as follows: wavelet kernel-based primal twin support vector machine has been introduced in Section 2; experimental analysis of economic development prediction method based on the wavelet kernel-based primal twin support vector machine algorithm is described in Section 3; and Section 4 gives the conclusions.

## 2. The Proposed Wavelet Kernel-Based Primal Twin Support Vector Machine

The wavelet kernel-based primal twin support vector machine algorithm can solve two smaller sized quadratic programming problems instead of solving a large one as in the traditional support vector machine algorithm. Given a set of training sets  $\{(x_i, y_i)\}_{i=1}^n \subset R^m \times R$ , where  $x_i$  and  $y_i$  denote the input vector and corresponding output, the two optimization problems of primal twin support vector machine can be modified as follows.

Minimize

$$T(w_1, \xi) = \left[ \frac{1}{2} (\|w_1\|_2^2 + b_1^2) + \frac{C_1}{2} \times \|f - (\theta(A)w_1 + eb_1)\|_2^2 + C_2 e' \xi \right] \quad (1)$$

subject to

$$Y - (\theta(A)w_1 + eb_1) \geq e\varepsilon_1 - \xi, \quad \xi \geq 0. \quad (2)$$

Minimize

$$T(w_2, \eta) = \left[ \frac{1}{2} (\|w_2\|_2^2 + b_2^2) + \frac{C_3}{2} \times \|h - (\theta(A)w_2 + eb_2)\|_2^2 + C_4 e' \eta \right] \quad (3)$$

subject to

$$(\theta(A)w_2 + eb_2) - Y \geq e\varepsilon_2 - \eta, \quad \eta \geq 0, \quad (4)$$

where  $C_i$ ,  $i = 1, 2, 3, 4$ , are the penalty factors;  $w_1$ ,  $w_2$  denote the weight vector;  $b_1$ ,  $b_2$  denote bias term;  $\xi$ ,  $\eta$  are the positive slack variables; and  $\theta(\cdot)$  is the mapping function.

Then, we can obtain primal twin support vector machine by minimizing the following equation:

$$F(\alpha) = \left[ \frac{1}{2} (\alpha' K \alpha + b_1^2) + \frac{C_1}{2} \times \|(K\alpha + eb_1) - f\|_2^2 + C_2 e' l \cdot (K\alpha + eb_1 - f, \mu_0) \right], \quad (5)$$

where  $K = K(A, A')$  and  $u_0$  is a given  $n$ -dimensional vector; and  $\alpha$  is the Lagrangian multiplier.

It is well known that primal twin support vector machine with appropriate structure is able to gain excellent nonlinear regression function. In the study, wavelet kernel function can be used to deal with input variables of primal twin support vector machine.

Here, Morlet wavelet function can be selected as the kernel function of the proposed primal twin support vector machine, which can be described as follows:

$$K(x, x') = \prod_{i=1}^m \cos\left(1.75 \times \frac{x - x'}{a_i}\right) \exp\left(-\frac{\|x - x'\|_2^2}{2a_i^2}\right), \quad (6)$$

where  $a_i$  are the coefficients of wavelet function.

It is sufficient to prove the inequality:

$$F[x](w) = (2\pi)^{-m/2} \int \exp(-j(w \cdot x)) k(x) dx \geq 0, \quad (7)$$

where

$$k(x) = \prod_{i=1}^m \cos\left(\frac{1.75x}{a_i}\right) \exp\left(-\frac{\|x\|_2^2}{2a_i^2}\right) \int \exp(-j(w \cdot x)) k(x) dx = \prod_{i=1}^m \frac{|a_i| \sqrt{2\pi}}{2} \left( \exp\left(-\frac{(1.75 + w_i a_i)^2}{2}\right) + \exp\left(-\frac{(1.75 - w_i a_i)^2}{2}\right) \right). \quad (8)$$

## 3. Experimental Analysis

Economic development data of Anhui province from 1992 to 2009 [19] are used to study the prediction performance of the wavelet kernel-based primal twin support vector machine

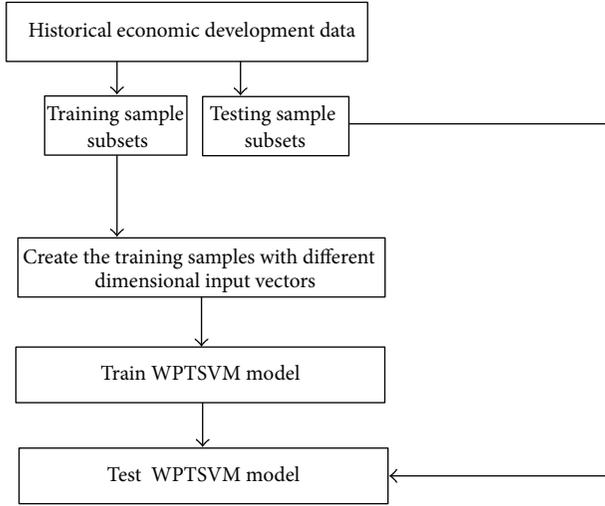


FIGURE 1: The experiment process of the wavelet kernel-based primal twin support vector machine algorithm.

TABLE 1: The comparison of mean error of economic development prediction values between WPTSVM and SVM.

The dimensions of input vector	Mean error of PTSVM	Mean error of SVM
3	0.0253	0.0410
4	0.0194	0.0336
5	0.0283	0.0462

algorithm. The experiment process of the wavelet kernel-based primal twin support vector machine algorithm can be shown in Figure 1; the training samples are created as follows:

$$\begin{aligned}
 X &= \begin{bmatrix} x_1 \\ x_2 \\ \dots \\ x_{n-m} \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & \dots & a_m \\ a_2 & a_3 & \dots & a_{m+1} \\ \dots & \dots & \dots & \dots \\ a_{n-m} & a_{n-m+1} & \dots & a_{n-1} \end{bmatrix}, \\
 Y &= \begin{bmatrix} y_1 \\ y_2 \\ \dots \\ y_{n-m} \end{bmatrix} = \begin{bmatrix} a_1 \\ a_2 \\ \dots \\ a_{n-m} \end{bmatrix},
 \end{aligned} \tag{9}$$

where  $m$  is the dimension of the input vector.

If the dimension of the input vector is set to 3, the training samples of this experiment can be described as follows:

$$\begin{aligned}
 X &= \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \dots \end{bmatrix} = \begin{bmatrix} d_{1992} & d_{1993} & d_{1994} \\ d_{1993} & d_{1994} & d_{1995} \\ d_{1994} & d_{1995} & d_{1996} \\ \dots & \dots & \dots \end{bmatrix}, \\
 Y &= \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \dots \end{bmatrix} = \begin{bmatrix} d_{1995} \\ d_{1996} \\ d_{1997} \\ \dots \end{bmatrix}.
 \end{aligned} \tag{10}$$

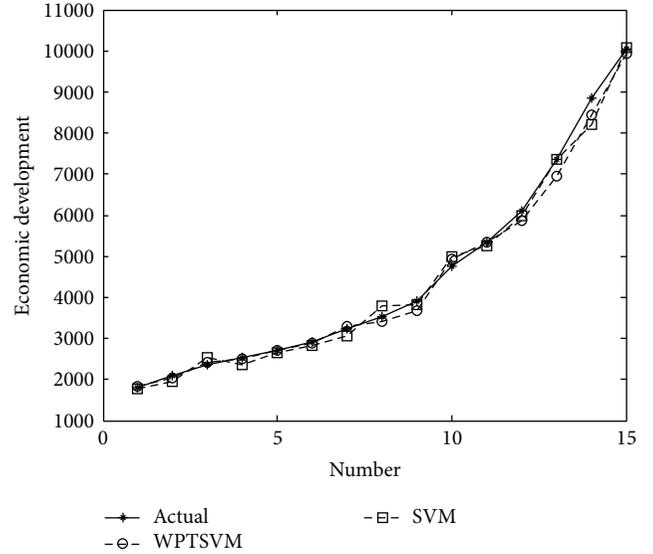


FIGURE 2: The comparison of the prediction values between the WPTSVM model and traditional support vector machine model trained by the training samples with 3-dimensional input vector, respectively.

In this experiment, we employ the training samples with different dimensional input vectors to train the wavelet kernel-based primal twin support vector machine algorithm.

Figure 2 gives the comparison of the prediction values between the wavelet kernel-based primal twin support vector machine model and traditional support vector machine model trained by the training samples with 3-dimensional input vector, respectively; and the comparison of the prediction error between the wavelet kernel-based primal twin support vector machine model and traditional support vector machine model trained by the training samples with 3-dimensional input vector, respectively is given in Figure 3. Then, the comparison of the prediction values between the wavelet kernel-based primal twin support vector machine model and traditional support vector machine model trained by the training samples with 4-dimensional input vector, respectively, is given in Figure 4; and Figure 5 gives the comparison of the prediction error between the wavelet kernel-based primal twin support vector machine model and traditional support vector machine model trained by the regression training samples with 4-dimensional input vector, respectively.

Finally, Figure 6 gives the comparison of the prediction values between the wavelet kernel-based primal twin support vector machine model and traditional support vector machine model trained by the training samples with 5-dimensional input vector, respectively; and the comparison of the prediction error between the wavelet kernel-based primal twin support vector machine model and traditional support vector machine model trained by the regression training samples with 5-dimensional input vector, respectively, is given in Figure 7. Table 1 gives the comparison of mean error

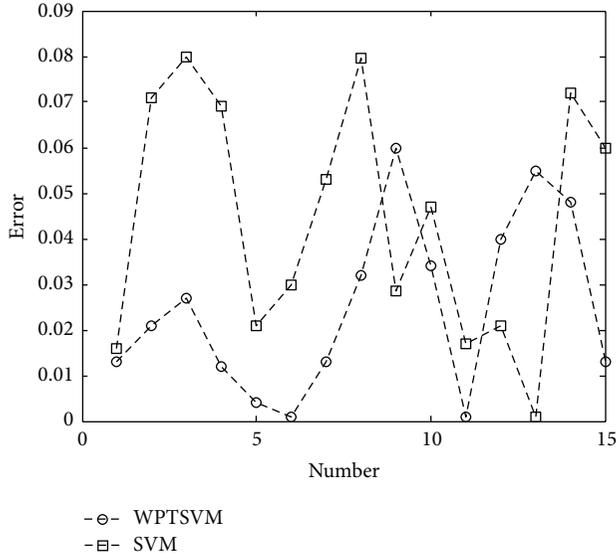


FIGURE 3: The comparison of the prediction error between the WPTSVM model and traditional support vector machine model trained by the training samples with 3-dimensional input vector, respectively.

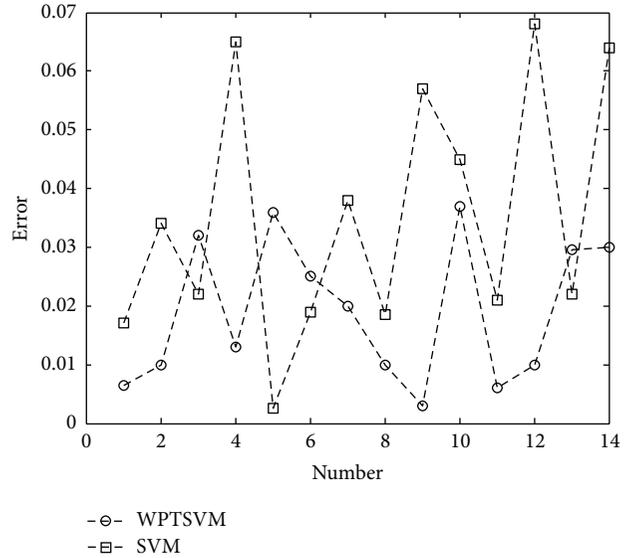


FIGURE 5: The comparison of the prediction error between the WPTSVM model and traditional support vector machine model trained by the regression training samples with 4-dimensional input vector, respectively.

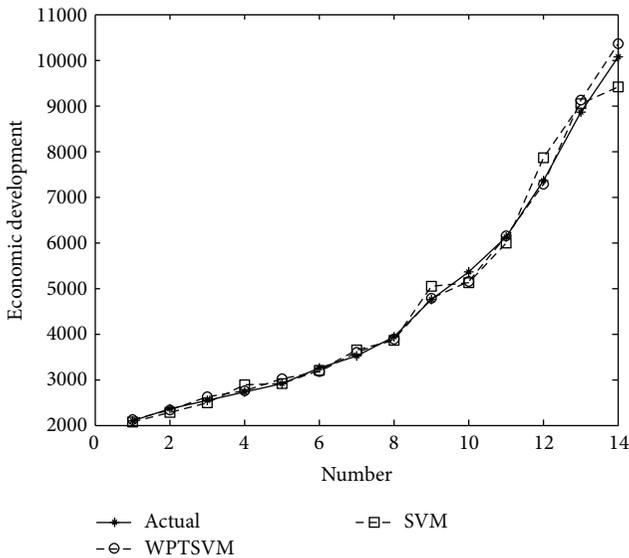


FIGURE 4: The comparison of the prediction values between the WPTSVM model and traditional support vector machine model trained by the training samples with 4-dimensional input vector, respectively.

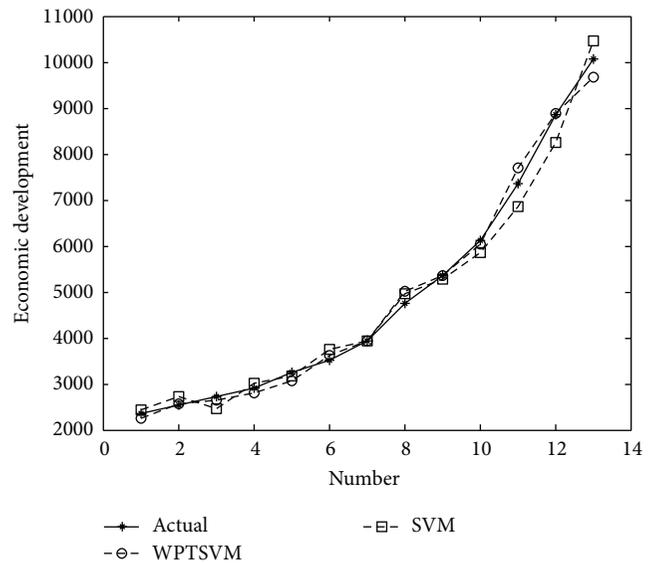


FIGURE 6: The comparison of the prediction values between the WPTSVM model and traditional support vector machine model trained by the training samples with 5-dimensional input vector, respectively.

of economic development prediction between wavelet kernel-based primal twin support vector machine and traditional support vector machine models trained by the training samples with the 3–5-dimensional input vectors, respectively. It can be seen that the economic development prediction accuracy of the wavelet kernel-based primal twin support vector machine model is better than that of traditional support vector machine.

#### 4. Conclusions

The wavelet kernel-based primal twin support vector machine algorithm is proposed to predict economic development in the paper. The WPTSVM algorithm can solve two smaller sized quadratic programming problems instead of solving a large one as in the traditional support vector machine algorithm. Morlet wavelet function is employed to construct the primal twin support vector machine.

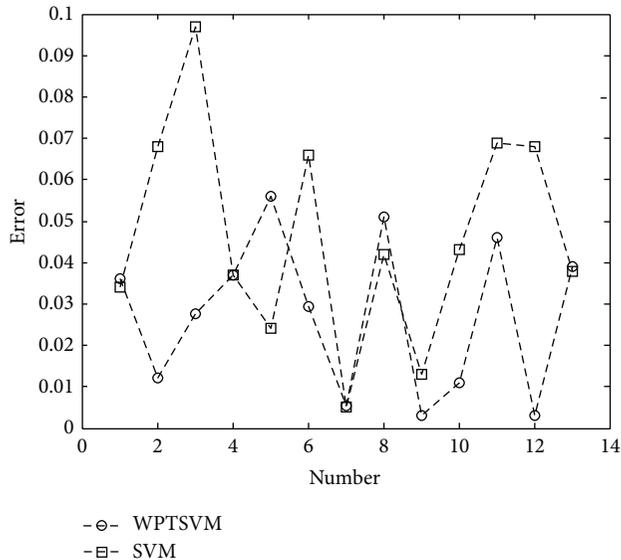


FIGURE 7: The comparison of the prediction error between the WPTSVM model and traditional support vector machine model trained by the regression training samples with 5-dimensional input vector, respectively.

Then, we use multistep prediction mode to indicate the practicability and stability of the wavelet kernel-based primal twin support vector machine algorithm. In this experiment, we employ the training samples with different dimensional input vectors to train the wavelet kernel-based primal twin support vector machine algorithm. The comparison of mean error of economic development prediction between wavelet kernel-based primal twin support vector machine and traditional support vector machine model trained by the training samples with the 3–5-dimensional input vectors, respectively is given in this paper. The testing results show that the economic development prediction accuracy of the WPTSVM model is better than that of traditional SVM.

## Acknowledgments

This work is supported by (1) Chinese Natural Science Foundation “Research on the Ecological Compensation Mechanism of the Inner River Basins of the Northwestern China—A Case Study of Shiyang River Basin (Grant no. 41171116)”; (2) General Research on Social Sciences of Ministry of Education of China (Grant no. 12YJAZH110).

## References

- [1] S. Anwar and A. Cooray, “Financial development, political rights, civil liberties and economic growth: evidence from South Asia,” *Economic Modelling*, vol. 29, no. 3, pp. 974–981, 2012.
- [2] Q. Liang and J. Z. Teng, “Financial development and economic growth: evidence from China,” *China Economic Review*, vol. 17, no. 4, pp. 395–411, 2006.
- [3] J. Xu, J. Zhao, Z. Song, and M. Liu, “Prediction of ultimate bearing capacity of Tubular T-joint under fire using artificial neural networks,” *Safety Science*, vol. 50, no. 7, pp. 1495–1501, 2012.
- [4] C. L. Wu, K. W. Chau, and C. Fan, “Prediction of rainfall time series using modular artificial neural networks coupled with data-preprocessing techniques,” *Journal of Hydrology*, vol. 389, no. 1–2, pp. 146–167, 2010.
- [5] G. D. Wu and S. L. Lo, “Effects of data normalization and inherent-factor on decision of optimal coagulant dosage in water treatment by artificial neural network,” *Expert Systems with Applications*, vol. 37, no. 7, pp. 4974–4983, 2010.
- [6] M. Bildirici, Ö. Ö. Ersin, and M. Kökdener, “Genetic structure, consanguineous marriages and economic development: panel cointegration and panel cointegration neural network analyses,” *Expert Systems with Applications*, vol. 38, no. 5, pp. 6153–6163, 2011.
- [7] T. Y. Kim, K. J. Oh, I. Sohn, and C. Hwang, “Usefulness of artificial neural networks for early warning system of economic crisis,” *Expert Systems with Applications*, vol. 26, no. 4, pp. 583–590, 2004.
- [8] M. Xu, T. C. Wong, and K. S. Chin, “Modeling daily patient arrivals at emergency department and quantifying the relative importance of contributing variables using artificial neural network,” *Decision Support Systems*, vol. 54, no. 3, pp. 1488–1498, 2013.
- [9] M. Bashiri and A. F. Geranmayeh, “Tuning the parameters of an artificial neural network using central composite design and genetic algorithm,” *Scientia Iranica*, vol. 18, no. 6, pp. 1600–1608, 2011.
- [10] J. Shu, Z. Zhang, I. Gonzalez, and R. Karoumi, “The application of a damage detection method using artificial neural network and train-induced vibrations on a simplified railway bridge model,” *Engineering Structures*, vol. 52, pp. 408–421, 2013.
- [11] R. Yu and M. Abdel-Aty, “Utilizing support vector machine in real-time crash risk evaluation,” *Accident Analysis and Prevention*, vol. 51, pp. 252–259, 2013.
- [12] A. Azadeh, M. Saberi, A. Kazem, V. Ebrahimipour, A. Nourmohammadzadeh, and Z. Saberi, “A flexible algorithm for fault diagnosis in a centrifugal pump with corrupted data and noise based on ANN and support vector machine with hyper-parameters optimization,” *Applied Soft Computing*, vol. 13, no. 3, pp. 1478–1485, 2013.
- [13] B. Biswal, M. K. Biswal, P. K. Dash, and S. Mishra, “Power quality event characterization using support vector machine and optimization using advanced immune algorithm,” *Neurocomputing*, vol. 103, pp. 75–86, 2013.
- [14] L. Han, L. Han, and H. Zhao, “Orthogonal support vector machine for credit scoring,” *Engineering Applications of Artificial Intelligence*, vol. 26, no. 2, pp. 848–862, 2013.
- [15] H. Muzhou and H. Xuli, “The multidimensional function approximation based on constructive wavelet RBF neural network,” *Applied Soft Computing Journal*, vol. 11, no. 2, pp. 2173–2177, 2011.
- [16] J. Rafiee, M. A. Rafiee, N. Prause, and M. P. Schoen, “Wavelet basis functions in biomedical signal processing,” *Expert Systems with Applications*, vol. 38, no. 5, pp. 6190–6201, 2011.
- [17] R. C. Guido, “A note on a practical relationship between filter coefficients and scaling and wavelet functions of discrete wavelet transforms,” *Applied Mathematics Letters*, vol. 24, no. 7, pp. 1257–1259, 2011.

- [18] J. Rafiee, M. A. Rafiee, and P. W. Tse, "Application of mother wavelet functions for automatic gear and bearing fault diagnosis," *Expert Systems with Applications*, vol. 37, no. 6, pp. 4568–4579, 2010.
- [19] B. Liu and G. Y. Yang, "Forecasting the economic development of Anhui based on combination forecasting method," *Journal of West Anhui University*, vol. 27, no. 5, pp. 48–51, 2011 (Chinese).

## Research Article

# Optimal Design of Signal Controlled Road Networks Using Differential Evolution Optimization Algorithm

**Huseyin Ceylan**

*Department of Civil Engineering, Engineering Faculty, Pamukkale University, 20070 Denizli, Turkey*

Correspondence should be addressed to Huseyin Ceylan; [hceylan@pau.edu.tr](mailto:hceylan@pau.edu.tr)

Received 4 April 2013; Revised 5 July 2013; Accepted 8 July 2013

Academic Editor: Vishal Bhatnagar

Copyright © 2013 Huseyin Ceylan. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This study proposes a traffic congestion minimization model in which the traffic signal setting optimization is performed through a combined simulation-optimization model. In this model, the TRANSYT traffic simulation software is combined with Differential Evolution (DE) optimization algorithm, which is based on the natural selection paradigm. In this context, the Equilibrium Network Design (EQND) problem is formulated as a bilevel programming problem in which the upper level is the minimization of the total network performance index. In the lower level, the traffic assignment problem, which represents the route choice behavior of the road users, is solved using the Path Flow Estimator (PFE) as a stochastic user equilibrium assessment. The solution of the bilevel EQND problem is carried out by the proposed Differential Evolution and TRANSYT with PFE, the so-called DETRANSPFE model, on a well-known signal controlled test network. Performance of the proposed model is compared to that of two previous works where the EQND problem has been solved by Genetic-Algorithms- (GAs-) and Harmony-Search- (HS-) based models. Results show that the DETRANSPFE model outperforms the GA- and HS-based models in terms of the network performance index and the computational time required.

## 1. Introduction

Configuring the traffic signal timings is a challenging problem in transportation engineering as it is important to minimize delays and total travel time in the road networks. Since the drivers' route choice behavior is taken into account in the User Equilibrium (UE) manner, this problem is called the Equilibrium Network Design (EQND) problem. Two common solution methods have been discussed for solving the EQND problem: Mutually Consistent (MC) and bilevel approaches. In those approaches, some procedures have been developed for determining the optimal signal timings. Allsop and Charlesworth [1] presented an MC calculation for the EQND problem. In their study, the equilibrium link flows and resulting signal settings are calculated, respectively, by solving each problem until the convergence criterion has been satisfied. However, the resulting MC signal settings and equilibrium link flows are nonoptimal in many cases [2, 3].

Due to the multiobjective structure of the EQND problem, it is usually modeled as a bilevel programming problem, where certain design parameters are optimized at the upper

level, while the lower level represents a UE problem [4–10]. Heydecker and Khoo [4] presented a Linear Constraint Approximation (LCA) to the UE constraint and solved the EQND problem. On the other hand, effectiveness of the Sensitivity Analysis (SA) of UE flows in particular design issues has been investigated in several studies [6, 11, 12]. The basic principle of the SA-based methods is searching for the optimal solution using the simplex method by formulating a linearized subproblem at the current signal settings. Because of the nonconvex solution space of the EQND problem, LCA- and SA-based algorithms may only produce locally optimal solutions [13]. Thus, these optimization techniques are not guaranteed to obtain the global optimum solutions. Over the past several decades, some heuristics based on natural phenomena such as Genetic Algorithms (GAs) [14], Simulated Annealing (SA) [15], and Differential Evolution (DE) [16] have been developed to overcome this problem. These methods can be applied to both continuous (differentiable) and discontinuous (nondifferentiable) objective functions without requiring an extensive formulation. Moreover, they do not require specialized starting points and they are rarely

trapped into a local optimum solution [17]. These advantages enhance their application in the field of the EQND.

Friesz et al. [18, 19] presented a metaheuristic SA-based optimization approach under variational inequality constraints. Ceylan and Bell [7] presented a simulation-optimization model combining GAs with the logit traffic assignment tool Path Flow Estimator (PFE), which has been developed and improved by Bell and Shield [20], Bell et al. [21–23], Bell and Iida [24], Bell and Grosso [25], and TRANSYT traffic simulation software [26, 27]. In the study, the upper level problem was considered to be the optimization of the green times, common cycle time, and offsets to minimize the network Performance Index (PI) which may be defined as the sum of a weighted combination of the delays and stops on all the links in a network. Furthermore, Teklu et al. [28] presented a GA-based signal timing optimization method that considers drivers rerouting. Ceylan [29] combined GAs with the TRANSYT Hill-Climbing optimization routine and proposed a hybrid global/local optimization method for determining optimal signal timings. In the related study, drivers rerouting was ignored. Ceylan and Ceylan [30] developed a hybrid solution approach, where the metaheuristic Harmony Search (HS) and TRANSYT Hill-Climbing optimization methods were combined, considering drivers' route choice behavior. The performance of their model was compared with that of the pure HS- and GA-based solution models. Although a number of solution methods were developed and improved for the solution of the EQND problem, it may be useful to develop new models based on different optimization algorithms. Among them, DE is an important optimization algorithm due to its powerful solution ability in finding the global optimum solution.

The DE algorithm, which is an improved version of GAs, is a quite simple evolutionary approach that is quite faster and robust at numerical optimization, and it is more likely to find the global optimum of an objective function [31]. However, to create new solution vectors, the DE uses nonuniform crossover and tournament selection operators, while the bitwise flipping approach is used in GAs. Thus, it can be used for optimizing functions, including many local optima with real variables [17]. The DE algorithm has recently been applied to several engineering design problems, including the optimization of the airfoil cross section of a vertical-axis wind turbine [32], the designing of the aperture of pyramidal horns [33], the optimizing of reservoir systems [34], shape design of variable capacitance micromotor [35], various constrained engineering design problems [36, 37], the vehicle routing problem with time windows [38], the optimal design of water distribution networks [39, 40], and river suspended sediment concentration modeling [41], but no studies of DE have been reported on the solution of the EQND problem.

In this study, the Differential Evolution and TRANSYT with PFE (DETRANSPFE) model are developed for the solution of the EQND problem. For this purpose, the Stochastic User Equilibrium (SUE) traffic assignment and traffic signal optimization problems are combined using the DE solution framework to minimize the network PI value. It should be

noted that this is the first time that the DE optimization algorithm is implemented for the EQND problem. The proposed model is applied to Allsop and Charlesworth's [1] test road network and a comparison with GA- and HS-based models is given to examine the solution ability of the DETRANSPFE model. The DETRANSPFE model further extends the GA- and HS-based models [7, 30] in terms of mathematical formulation and the computational time.

The remainder of the paper is organized as follows: fundamentals of the SUE assignment and traffic signal optimization problems are given in Section 2. Basics of the DE algorithm and DETRANSPFE model are provided in Section 3. Performance of DETRANSPFE model is proved with a numerical example in Section 4. Last, the study is ended with some conclusions and future recommendations in Section 5.

## 2. Problem Formulation

Considering a network with sets of nodes  $N$ , links  $A$ , and paths  $P$  between the Origin-Destination ( $O$ - $D$ ) pairs  $W$ , bilevel formulation of the EQND problem may be expressed as a combination of the SUE traffic assignment and network performance optimization in the following way:

$$\begin{aligned} \min \quad & \text{PI}(\mathbf{v}^*(\boldsymbol{\psi}), \boldsymbol{\psi}) \\ & = \left\{ \sum_{a \in A} [G \cdot d \cdot D_a(\mathbf{v}^*(\boldsymbol{\psi}), \boldsymbol{\psi}) \right. \\ & \quad \left. + K \cdot k \cdot L_a(\mathbf{v}^*(\boldsymbol{\psi}), \boldsymbol{\psi}) \right\} \\ & + \lambda \cdot E(\phi) \end{aligned} \quad (1a)$$

$$\text{s.t.} \quad C_{\min} \leq C \leq C_{\max} \quad (1b)$$

$$\phi \geq \phi_{\min}, \quad \phi \in \boldsymbol{\varphi} \quad (1c)$$

$$E(\phi) = \begin{cases} \left| \left[ \sum_{m_n} (\phi + I) \right] - C \right| & \text{if } \sum_{m_n} (\phi + I) \neq C, [\forall n \in N] \\ 0 & \text{if } \sum_{m_n} (\phi + I) = C, [\forall n \in N], \end{cases} \quad (1d)$$

where  $\text{PI}(\mathbf{v}^*(\boldsymbol{\psi}), \boldsymbol{\psi})$  is the performance index of the road network ( $\text{£/h}$ ),  $\mathbf{v}^*(\boldsymbol{\psi})$  is the vector of equilibrium link flows as a function of signal timings,  $\boldsymbol{\psi}$ .  $G$  is the cost of delay,  $d$  is the weighting factor of delay,  $D_a(\mathbf{v}^*(\boldsymbol{\psi}), \boldsymbol{\psi})$  represents delay as a function of equilibrium link flows and signal timings,  $K$  is the cost for 100 vehicle stops,  $k$  is the weighting factor of stops, and  $L_a(\mathbf{v}^*(\boldsymbol{\psi}), \boldsymbol{\psi})$  is the number of vehicle stops per hour. Additionally, subscripts " $a$ ," " $n$ ," and " $p$ " denote link  $a$ , node (or signalized intersection)  $n$ , and path  $p$ , respectively. Superscript "\*" indicates that the variable is associated with its optimal and/or equilibrium state.  $E(\phi)$  is the penalty term, and it takes a zero value if the total duration of the total green and intergreen times equals the network cycle time at every signalized intersection; otherwise, it varies linearly with

*Step 1* (initialization)  
 $\mathbf{q} \leftarrow$  user supplied,  $\forall w \in W$   
 $v_a \leftarrow 0, \forall a \in A$   
 $t_a \leftarrow t_a(v_a), \forall a \in A$   
 $j \leftarrow 1$ , iteration counter

*Step 2* (update link costs by method of successive averages)  
 $t_a \leftarrow \frac{1}{j}t_a(v_a) + \left(1 - \frac{1}{j}\right)t_a$

*Step 3* (new path costs)  
 $y_p \leftarrow \sum_{a \in A} \delta_{ap} t_a(v_a)$

*Step 4* (new path flows - logit route choice model)  
 $h_p \leftarrow \frac{q_w \exp(-\alpha y_p)}{\sum_{p \in P_w} \exp(-\alpha y_p)}$

*Step 5* (new link flows)  
 $v_a \leftarrow \sum_{p \in P_w} \delta_{ap} h_p$   
 if any new path and link flows converged then  
 $j \leftarrow j + 1$   
 return to Step 2  
 else stop.

PSEUDOCODE 1

the magnitude of constraint violation. The penalty approach is implemented by multiplying the penalty term with a weighting factor,  $\lambda$ , as shown in (1a). Note that a larger  $\lambda$  for a given constraint set means that a greater emphasis will be placed on resolving constraint violations. However, the selection of  $\lambda$  is mostly problem dependent. Furthermore,  $C$  represents the network cycle time while  $C_{\min}$  and  $C_{\max}$  are the lower and upper bounds of the network cycle time, respectively,  $\phi_{\min}$  is the lowest acceptable stage green timing duration,  $\boldsymbol{\phi}$  is the vector of duration of stage green timings [ $\phi; \forall \phi = 1, \dots, m_n, \forall n \in N$ ],  $m_n$  is the number of stages for signalized intersection  $n$ , and  $I$  is the intergreen time between signal stages.

In (1a),  $\mathbf{v}^*(\boldsymbol{\psi})$  may implicitly be derived from the solution of (2a), which leads to a logit path choice model [42]

$$\begin{aligned} \text{Minimize } Z(\mathbf{v}, \boldsymbol{\psi}) \\ = -\mathbf{q}^T \mathbf{y}(\mathbf{v}, \boldsymbol{\psi}) + \mathbf{v}^T \mathbf{t}(\mathbf{v}, \boldsymbol{\psi}) \\ - \sum_{a \in A} \int_0^{v_a(\boldsymbol{\psi})} t_a(\boldsymbol{\psi}, x) dx \end{aligned} \quad (2a)$$

$$\text{s.t. } \mathbf{q} = \boldsymbol{\Lambda} \mathbf{h} \quad (2b)$$

$$\mathbf{v} = \boldsymbol{\delta} \mathbf{h} \quad (2c)$$

$$\mathbf{h} \geq \mathbf{0}, \quad (2d)$$

where  $\mathbf{q}$  represents the vector of the total  $O$ - $D$  demands [ $q_w; \forall w \in W$ ],  $\mathbf{y}(\mathbf{v}, \boldsymbol{\psi})$  is the vector of path travel times [ $y_p; \forall p \in P$ ],  $\mathbf{t}(\mathbf{v}, \boldsymbol{\psi})$  is the vector of link travel times [ $t_a; \forall a \in A$ ], and  $\mathbf{h}$  is the vector of path flows [ $h_p; \forall p \in P$ ]. In (2b) and (2c),  $\boldsymbol{\Lambda}$  represents the  $O$ - $D$ /path incidence matrix and  $\boldsymbol{\delta}$  represents the link/path incidence matrix, where  $\delta_{ap} = 1$  if

link  $a$  is on path  $p$ , and  $\delta_{ap} = 0$ ; otherwise, [ $\delta_{ap}; \forall a \in A; \forall p \in P$ ]. In this study, solution of the SUE assignment problem, which is given in (2a), is conducted by PFE with Pseudocode 1, where  $\alpha$  is the dispersion parameter. When  $\alpha$  is zero, path choice is not sensitive to cost that means that any path is likely to be chosen. On the other hand, while  $\alpha$  tends to infinity, drivers become concentrated on the least cost paths to travel. The link cost function used by PFE is presented in

$$t_a(v_a(\boldsymbol{\psi}), \boldsymbol{\psi}) = t_a^0 + d_a^u + d_a^{ro}(t), \quad (3)$$

where  $t_a^0$  is the free-flow travel time on link  $a$ ,  $d_a^u$  and  $d_a^{ro}$  are the uniform and random plus over saturation delays at the downstream intersection, respectively. The convergence measure is obtained by basing it on the flow in the last several iterations of PFE. At this point, let  $\bar{v}_a^j$  represent the average flow over the last  $r$  iterations; that is,

$$\bar{v}_a^j = \frac{1}{r} (v_a^j + v_a^{j-1} + \dots + v_a^{j-r+1}) = \frac{1}{r} \sum_{i=0}^{r-1} v_a^{j-i}, \quad (4)$$

where  $r$  is a priori fixed that may be accepted as “3” in applications [43]. Thus, a convergence criterion  $\kappa$ , that is based on flow similarity, is used as given in

$$\frac{\sqrt{\sum_a (v_a^{j+1} - v_a^j)^2}}{\sum_a v_a^j} \leq \kappa. \quad (5)$$

### 3. Basics of the Differential Evolution Algorithm and DETRANSPFE Model

The DE algorithm is a comparatively simple, fast, powerful, and robust variant of an evolutionary algorithm for solving

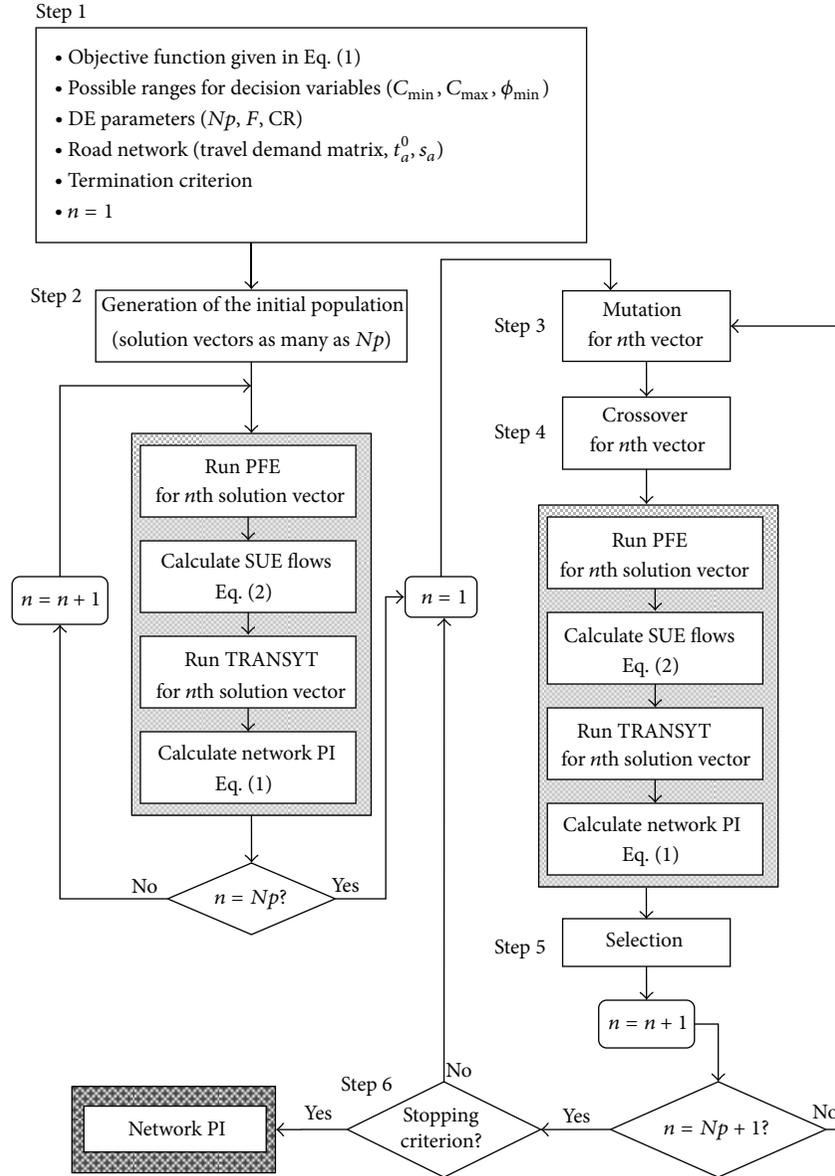


FIGURE 1: Flow chart of the DETRANSPFE model.

optimization problems. In this section, the solution of the EQND problem, which deals with the signal setting optimization based on the DETRANSPFE model, is presented. The solution procedure of the proposed model is presented in Figure 1 and the main solution steps are explained hereafter.

*Step 1* (initialization of the objective function, user-specified DE parameters, network characteristics, and termination criterion). The objective function to be minimized has been given in (1a). The DE algorithm is controlled by three parameters. The first parameter, which is called the number of populations ( $Np$ ), that represents the number of solution vectors used during the optimization process. The mutation factor ( $F$ ), which is the second parameter and which is recommended to be set between 0.5 and 1 by Storn and

Price [31], has an effect on the difference vector. The third one is the crossover rate (CR), which is the probability of mixing parameters of the mutation vector. The recommended range of the crossover rate is [0.8, 1.0] by Storn and Price [31]. Fixed sets of data related to the network topology are represented with the vector of free-flow link travel times,  $\mathbf{t}^0 [t_a^0; \forall a \in A]$ , and the vector of saturation link flows,  $\mathbf{s} [s_a; \forall a \in A]$ . In this study, results of the DETRANSPFE model will be compared with those obtained by the GA- and HS-based models, which have been developed by Ceylan and Bell [7] and Ceylan and Ceylan [30], respectively. Therefore, the maximum number of generations ( $\xi_{\max}$ ) has been used as the termination criterion as in GA- and HS-based models. The numerical values of the DE parameters, the network characteristics, and the termination criterion are given in Section 4.

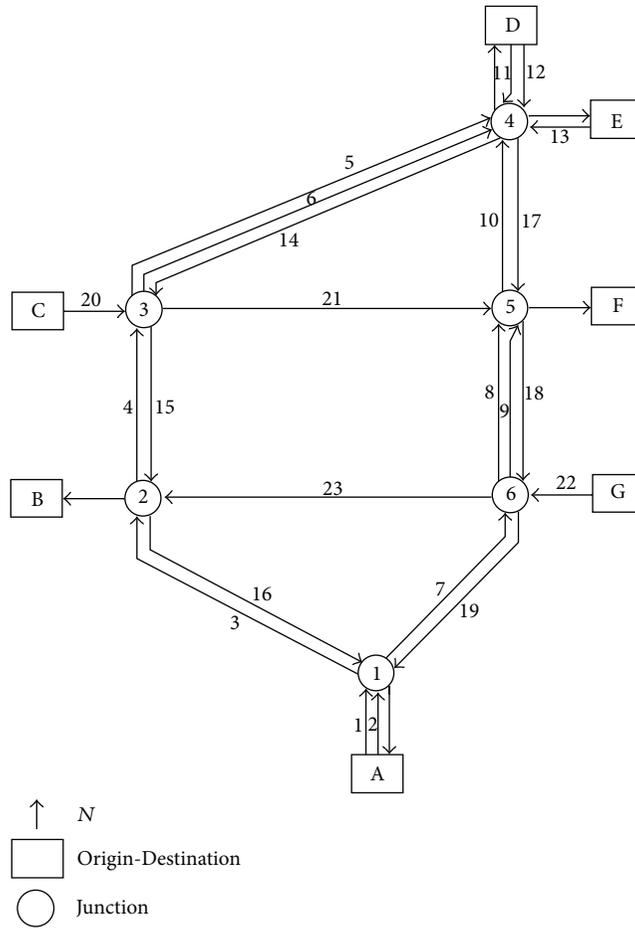


FIGURE 2: Allsop and Charlesworth's [1] test network.

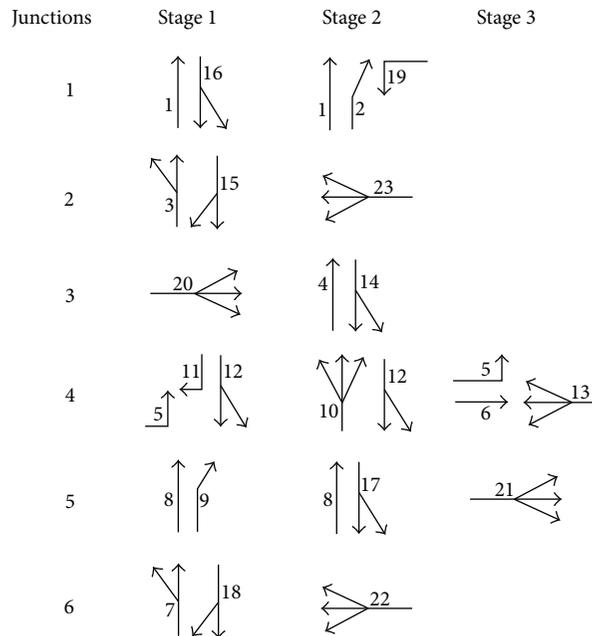


FIGURE 3: Configuration of signal groups.

*Step 2* (generation of the initial population). In this step, initial signal timing variables are generated between preset possible bounds and their corresponding network PI values which are calculated and stored as shown in

$$\begin{aligned} & \begin{bmatrix} C_1^1 & \phi_2^1 & \cdots & \theta_{\eta-1}^1 & \theta_{\eta}^1 \\ C_1^2 & \phi_2^2 & \cdots & \theta_{\eta-1}^2 & \theta_{\eta}^2 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ C_1^{Np-1} & \phi_2^{Np-1} & \cdots & \theta_{\eta-1}^{Np-1} & \theta_{\eta}^{Np-1} \\ C_1^{Np} & \phi_2^{Np} & \cdots & \theta_{\eta-1}^{Np} & \theta_{\eta}^{Np} \end{bmatrix} \\ & \Rightarrow \begin{bmatrix} \text{PI}(\mathbf{v}, \boldsymbol{\psi})^1 \\ \text{PI}(\mathbf{v}, \boldsymbol{\psi})^2 \\ \cdots \\ \text{PI}(\mathbf{v}, \boldsymbol{\psi})^{Np-1} \\ \text{PI}(\mathbf{v}, \boldsymbol{\psi})^{Np} \end{bmatrix}, \end{aligned} \quad (6)$$

where  $\eta$  is the number of decision variables and  $\boldsymbol{\theta}$  represents the vector of offset variables [ $\theta; \forall \theta = 1, \dots, N$ ]. In order to calculate the network PI values according to (1a), the TRANSYT traffic simulation tool, which requires the SUE equilibrium link flows, is used at this step. As can be seen from Figure 1, TRANSYT requires both signal timing variables and corresponding SUE traffic link flows to calculate the network PI. Thus, before running the TRANSYT, the PFE traffic assignment tool is employed to calculate the SUE traffic link flows. Signal timing variables are transformed for the use of DETRANSFPE model as follows.

- (i) Network cycle time is generated based on possible value bound [ $C_{\min}, C_{\max}$ ].
- (ii) Offsets are generated based on the following expression:

$$0 \leq \theta_i \leq C - 1, \quad (7)$$

where  $\theta_i$  represents the offset variable for signalized intersection  $i$  [ $\theta_i \in \boldsymbol{\theta}; i = 1, 2, \dots, N$ ]. However, stage change times are derived from offset variables in the following way:

$$S_{i,1} = \theta_i, \quad i = 1, 2, \dots, N, \quad (8)$$

where  $S_{i,1}$  is the change time for the first stage of the  $i$ th signalized intersection [44].

- (iii) Stage green times are generated based on the possible value bound [ $\phi_{\min}, \phi_{\max}$ ].

Note that, unlike the DE and HS in which the signal timing parameters governing the system performance index are represented as real numbers, these parameters in the GA are coded into binary strings, and the mapping from the binary string representation of variables into the real numbers is carried out to be used in the TRANSYT traffic model and PFE [7]. At this point, simplicity to code is important for researchers who are not familiar with programming and are looking for algorithms that can be simply implemented and tuned to solve the traffic signal timing optimization problem.

TABLE 1: Fixed travel demands for each *O-D* pair (vehicles/hour).

<i>O-D</i> pairs	F	E	D	B	A	Origin totals
A	200	30	700	250	—	1180
C	900	130	200	20	40	1290
D	100	50	—	250	400	800
E	20	—	30	130	300	480
G	20	60	170	450	550	1250
Destination totals	1240	270	1100	1100	1290	5000

TABLE 2: Input data for test network.

Intersection	Link	$t_a^0$	$s_a$
1	1	1	2000
	2	1	1600
	16	10	2900
	19	10	1500
2	3	10	3200
	15	15	2600
	23	15	3200
3	4	15	3200
	14	20	3200
	20	1	2800
	5	20	1800
4	6	20	1850
	10	10	2200
	11	1	2000
	12	1	1800
	13	1	2200
5	8	15	1850
	9	15	1700
	17	10	1700
6	21	15	3200
	7	10	1800
	18	15	1700
	22	1	3600

*Step 3* (mutation). In particular, DE has the advantage of using a simple and efficient form of a self-adapting mutation process [45]. The mutation, which represents the basic strength of DE, is performed by adding the weighted difference vector between two population members to a third member. Combining three different, randomly chosen signal timing vectors to create a mutant vector,  $\boldsymbol{\rho}_{i,g}$ , is given in

$$\boldsymbol{\rho}_{i,G} = \boldsymbol{\psi}_{r0,G} + F \cdot (\boldsymbol{\psi}_{r1,G} - \boldsymbol{\psi}_{r2,G}), \quad (9)$$

where  $r0$  is a randomly chosen base vector index that is different from the target vector index,  $i$ .  $r1$  and  $r2$  are also randomly selected difference vector indices that are different from both base and target vector indices. Simple adaptation rule for  $F$  improves the performance of the DE algorithm to a large extent without imposing any serious computational burden [46]. However, the mutation is treated as a random change of some signal timing parameters in the GA, and this

process may lead to an increase in computational load since the mutation is repeated if the mutated solution is infeasible.

*Step 4* (crossover). The search process of the DE is completed with the uniform crossover. At this step, each member of the trial vector,  $\mu_{i,g}$ , is chosen from the mutant vector with the probability of CR or from the target vector with the probability of 1-CR as given in

$$\mu_{j,i,G} = \begin{cases} \rho_{j,i,G} & \text{if } (\text{rand}_j(0,1) \leq \text{CR or } j = j_{\text{rand}}) \\ \psi_{j,i,G} & \text{otherwise.} \end{cases} \quad (10)$$

CR is compared with the output of a uniform random number generator  $\text{rand}_j(0,1)$  to determine whether the mutant vector or target vector will provide the member of the trial vector in (10). If the random number is less than or equal to CR, the trial parameter is chosen from the mutant vector,  $\rho_{i,G}$ ; otherwise, the parameter is chosen from the target vector,  $\psi_{i,G}$ . Additionally, the condition of  $j = j_{\text{rand}}$  ensures that at least one member of the trial vector is inherited from the mutant vector. Although both DE and GA contain the crossover process, their implementations are quite different. In DE, one of two feasible signal timing parameter values is chosen, whereas two solution vectors are recombined to produce two trial vectors often by crossover in GA. Considering the possible bounds of the signal timing parameters, the crossover operator of the GA used to manipulate the chromosomes often yields infeasible offspring. In such cases, crossover process is repeated and valuable CPU time is wasted.

*Step 5* (selection). At this step, the objective function is evaluated using the parameters of the trial vector. For this purpose, DETRANSPFE employs PFE for calculating the SUE link flows using signal timing parameters stored in the trial vector. Then, both trial vector parameters and resulting SUE link flows are given to the TRANSYT to calculate the total network PI. If the trial vector,  $\mu_{i,g}$ , has an equal or lower PI value than that of its target vector,  $\psi_{i,G}$ , it replaces the target vector in the next generation; otherwise, the target vector is retained in the population as given in

$$\psi_{i,G+1} = \begin{cases} \mu_{i,G} & \text{if } \text{PI}(\mathbf{v}^*(\mu_{i,G}), \mu_{i,G}) \leq \text{PI}(\mathbf{v}^*(\psi_{i,G}), \psi_{i,G}) \\ \psi_{i,G} & \text{otherwise.} \end{cases} \quad (11)$$

It should be noted that although the mutation, crossover, and selection steps are explained only for stage green timing parameters in (9)–(11), these steps are performed in the same manner for all decision parameters in the DETRANSPFE model.

*Step 6* (termination). The mutation, crossover, and selection steps of the DETRANSPFE model are repeated until the maximum number of generations,  $\xi_{\text{max}}$ , is reached.

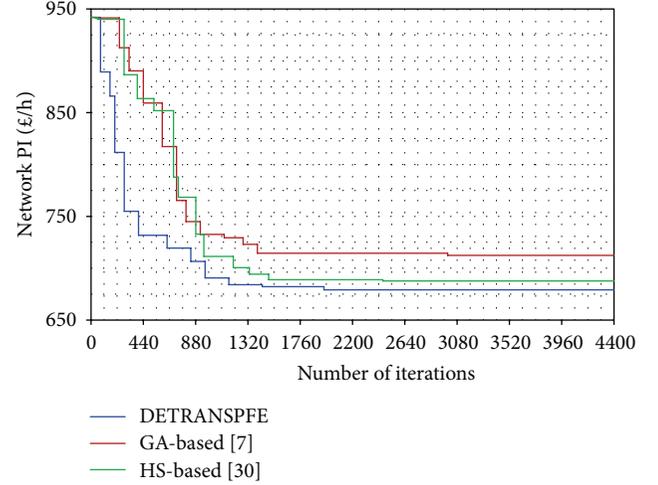


FIGURE 4: GA-based [7], HS-based [30], and DE-based model solutions of Allsop and Charlesworth's network.

TABLE 3: Numerical results of three models.

Models	Best PI (£/h)	Mean PI (£/h)	Worst PI (£/h)
GA-based [7]	712.5	N/A	N/A
HS-based [30]	687.7	690.4	704.4
DETRANSPFE	679.1	684.7	690.6

N/A: not available.

## 4. Numerical Example and Computational Comparison

The performance of the proposed DETRANSPFE model is investigated on Allsop and Charlesworth's [1] well-known test road network in this section. The results obtained by two previous works are presented to make a comparison to prove the effectiveness of the DETRANSPFE model.

*4.1. Test Road Network.* The basic layout of the test network and the allocations for signal groups for each intersection are provided in Figures 2 and 3, respectively.

Fixed travel demands for each *O-D* pair are given in Table 1. The input data including free-flow travel times and saturation flow rates is given in Table 2.

Using typical values found in practice, the minimum green time,  $\phi_{\text{min}}$ , for each group is 7 seconds, and the intergreen times,  $I$ , are 5 seconds between incompatible signal groups. The possible range for the network cycle time is set at 36 and 120 seconds. Values of the mutation factor,  $F$ , the crossover rate, CR, which are the user-specified parameters of DE, and the weighting factor,  $\lambda$ , are set to 0.80, 0.80, and 10, respectively. The optimization process is terminated after 110 generations for  $Np = 40$  (i.e., 4400 function evaluations) in the same manner as Ceylan and Bell's [7] GA-based modeling.

*4.2. Computational Results.* In order to prove the effectiveness of DETRANSPFE, the model was run 100 times with

TABLE 4: Signal timing values resulting from three models.

Models	Performance Index, PI (£/h)	Cycle time, C (sec)	Junction number, $n$	Start of green in seconds		
				Stage 1	Stage 2	Stage 3
GA-based [7]	712.5	77	1	0	32	—
			2	59	25	—
			3	13	60	—
			4	44	72	20
			5	64	5	30
			6	47	6	—
HS-based [30]	687.7	71	1	0	28	—
			2	50	20	—
			3	8	52	—
			4	37	63	15
			5	55	67	21
			6	37	69	—
DETRANSPFE	679.1	79	1	0	18	—
			2	36	11	—
			3	77	47	—
			4	38	67	15
			5	57	70	19
			6	22	52	—

different random seeds and initial populations for the test network. The convergence behavior of the DETRANSPFE, GA-based, and HS-based models can be seen in Figure 4.

Figure 4 indicates that the best solution was obtained as 679.1 £/h at the 49th generation (i.e., about 1690 function evaluations) with the DETRANSPFE model. The improvement of the network PI is about 28% in comparison with the initial status of the network. It can also be seen from Figure 4 that the DETRANSPFE model performs slightly better than the GA- and HS-based models in terms of the final values of total network performance index. The comparison of GA-based, HS-based, and DETRANSPFE models in terms of the network PI value is given in Table 3.

It can be seen from Table 3 that the DETRANSPFE model minimizes the network PI value by about 4.69% and 1.25% better than the GA- and HS-based models, respectively. Furthermore, the highest PI value obtained after 100 runs of the DETRANSPFE model is 690.6 £/h, which may indicate the better solution ability of the DETRANSPFE model. In order to evaluate the average computation times required by three models, the algorithms have been recoded in Visual Basic and run in a PC with a 1.6 Ghz processor and 1 GB ram. Computation time required by the DETRANSPFE model is obtained as 36 minutes, while GA- and HS-based models require about 66 and 43 minutes, respectively. As it has been mentioned in the previous section, the crossover and mutation operators of the GA may lead to an increase in computation time due to the infeasible solutions. Ceylan and Ceylan [30] stated that if signal timing constraints are not satisfied for generated timings, the HS-based model automatically discards those generated signal timings and starts generating new ones. Considering this situation, it may be concluded that the DETRANSPFE completes 4400 function

evaluations by about 45% and 16% faster than the GA- and HS-based models, respectively. The network cycle times obtained by the GA-based, HS-based, and DETRANSPFE models is 77, 71, and 79 seconds, respectively. Signal timings resulting from three models are given in Table 4.

Resulting link flows and degree of saturation values resulting from three models are presented in Tables 5(a) and 5(b), respectively.

Table 5(b) indicates that since the highest degree of saturation, is obtained as 86%, which is less than 90%, the network works under reserve capacity, and traffic flow is uncongested after the application of the DETRANSPFE model.

## 5. Conclusions

A number of methods for the solution of EQND problems have been discussed and new methodologies have been created. In this context, a simulation-optimization based-model, so-called DETRANSPFE, has been developed by combining the DE optimization technique with the TRANSYT traffic simulation tool. The EQND problem, which has been formulated in the bilevel form, has been solved by searching for the optimal or near-optimal signal setting strategy on the upper level with the DE optimization technique. On the lower level, the SUE assignment problem has been solved using PFE. In order to prove the effectiveness of the DETRANSPFE model, it has been applied to a well-known test network. Results have shown that the DETRANSPFE model provides better results than those obtained by GA- and HS-based models in terms of the total network performance index. In order to evaluate the average computation times required by three models, GA- and HS-based algorithms have been

TABLE 5: (a) Equilibrium link flows resulting from three models, (b) degree of saturation values resulting from three models.

(a)			
Final values of SUE link flows (veh/h)			
Links	GA-based [7]	HS-based [30]	DETRANSPFE
1	716	718	769
2	463	462	411
3	716	718	769
4	569	594	528
5	636	638	592
6	173	174	146
7	462	461	411
8	478	477	552
9	120	110	99
10	479	479	553
11	499	498	500
12	250	252	250
13	450	450	450
14	789	789	790
15	790	788	791
16	663	662	391
17	409	411	410
18	350	352	349
19	625	629	899
20	1290	1290	1290
21	1057	1069	1079
22	1250	1250	1250
23	837	847	460

(b)			
Degree of saturation (%)			
Links	GA-based [7]	HS-based [30]	DETRANSPFE
1	36	36	38
2	54	53	36
3	44	43	38
4	54	57	50
5	57	57	54
6	34	35	31
7	62	65	69
8	51	56	64
9	39	57	51
10	80	81	86
11	80	80	79
12	22	23	22
13	75	76	81
14	73	76	75
15	60	58	48
16	63	68	76
17	88	82	79
18	50	52	62
19	78	76	83
20	82	82	81
21	85	79	78
22	72	70	61
23	67	72	54

recoded and run. Results indicated that the DETRANSPFE model requires about 45% and 16% less computational time in comparison with the GA- and HS-based models, respectively.

Several directions, for future research in solving the EQND problem, are possible. For DE, encouraging results indicate that there is a potential for further improvement in its procedures, such as developing a hybrid search mechanism, in which a local search technique is integrated with DE algorithm. Besides, the design of different metaheuristics, which may improve the solution quality in terms of the system performance index and computational time required, may be explored. For practical application purposes, it would be useful to carry out a comparative study based on different metaheuristics within the scope of a case study including a real-sized signal controlled road network.

**Conflict of Interests**

The author declares that there is no conflict of interests.

**References**

- [1] R. E. Allsop and J. A. Charlesworth, "Traffic in a signal-controlled road network: an example of different signal timings including different routings," *Traffic Engineering and Control*, vol. 18, no. 5, pp. 262–265, 1977.
- [2] S. B. Gershwin and H. N. Tan, "Hybrid optimisation: optimal static traffic control constrained by drivers' route choice behaviour," Laboratory for Information and Decision System Report, Massachusetts Institute of Technology, 1979.
- [3] T. J. Dickson, "A note on traffic assignment and signal timings in a signal-controlled road network," *Transportation Research Part B*, vol. 15, no. 4, pp. 267–271, 1981.
- [4] B. G. Heydecker and T. K. Khoo, "The equilibrium network design problem," in *Proceedings of the Conference on Models and Methods for Decision Support (AIRO '90)*, pp. 587–602, Sorrento, Italy, 1990.
- [5] S. Suh and T. J. Kim, "Solving nonlinear bilevel programming models of the equilibrium network design problem: a comparative review," *Annals of Operations Research*, vol. 34, no. 1–4, pp. 203–218, 1992.
- [6] H. Yang and S. Yagar, "Traffic assignment and signal control in saturated road networks," *Transportation Research Part A*, vol. 29, no. 2, pp. 125–139, 1995.
- [7] H. Ceylan and M. G. H. Bell, "Traffic signal timing optimisation based on genetic algorithm approach, including drivers' routing," *Transportation Research Part B*, vol. 38, no. 4, pp. 329–342, 2004.
- [8] H. Ceylan and M. G. H. Bell, "Genetic algorithm solution for the stochastic equilibrium transportation networks under congestion," *Transportation Research B*, vol. 39, no. 2, pp. 169–185, 2005.
- [9] S.-W. Chiou, "A generalized iterative scheme for network design problem," *Applied Mathematics and Computation*, vol. 188, no. 2, pp. 1115–1123, 2007.
- [10] S.-W. Chiou, "A hybrid approach for optimal design of signalized road network," *Applied Mathematical Modelling*, vol. 32, no. 2, pp. 195–207, 2008.
- [11] R. L. Tobin and T. L. Friesz, "Sensitivity analysis for equilibrium network flow," *Transportation Science*, vol. 22, no. 4, pp. 242–250, 1988.
- [12] T. L. Friesz, R. L. Tobin, H.-J. Cho, and N. J. Mehta, "Sensitivity analysis based heuristic algorithms for mathematical programs

- with variational inequality constraints," *Mathematical Programming*, vol. 48, no. 1–3, pp. 265–284, 1990.
- [13] S.-W. Chiou, "An efficient algorithm for optimal design of area traffic control with network flows," *Applied Mathematical Modelling. Simulation and Computation for Engineering and Environmental Systems*, vol. 33, no. 6, pp. 2710–2722, 2009.
- [14] D. E. Goldberg, *Genetic Algorithms in Search Optimization and Machine Learning*, Addison-Wesley, Reading, Mass, USA, 1989.
- [15] S. Kirkpatrick, C. D. Gelatt, Jr., and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, 1983.
- [16] R. Storn and K. Price, "Differential evolution: a simple and efficient adaptive scheme for global optimization over continuous spaces," Tech. Rep. TR-95-012, ICSI, USA, 1995.
- [17] M. H. Khademi, P. Setoodeh, M. R. Rahimpour, and A. Jahanmiri, "Optimization of methanol synthesis and cyclohexane dehydrogenation in a thermally coupled reactor using differential evolution (DE) method," *International Journal of Hydrogen Energy*, vol. 34, no. 16, pp. 6930–6944, 2009.
- [18] T. L. Friesz, H.-J. Cho, N. J. Mehta, R. L. Tobin, and G. Anandalingam, "Simulated annealing approach to the network design problem with variational inequality constraints," *Transportation Science*, vol. 26, no. 1, pp. 18–26, 1992.
- [19] T. L. Friesz, G. Anandalingam, N. J. Mehta, K. Nam, S. J. Shah, and R. L. Tobin, "The multiobjective equilibrium network design problem revisited: a simulated annealing approach," *European Journal of Operational Research*, vol. 65, no. 1, pp. 44–57, 1993.
- [20] M. G. H. Bell and C. M. Shield, "Log-linear model for path flow estimation," in *Proceedings of the 4th International Conference on Applications of Advanced Technologies in Transportation Engineering*, pp. 695–699, June 1995.
- [21] M. G. H. Bell, W. H. K. Lam, and Y. A. Iida, "A time-dependent multiclass path flow estimator," in *Proceedings of the 13th International Symposium on Transportation and Traffic Theory*, pp. 173–194, Pergamon Press, Lyon, France, 1996.
- [22] M. G. H. Bell, C. M. Shield, J. J. Henry, and L. Breheret, "A stochastic user equilibrium (SUE) path flow estimator for the DEDALE database in Lyon," in *Proceedings of the Advanced Methods in Transportation Analysis*, pp. 75–92, Springer, Berlin, Germany, 1996.
- [23] M. G. H. Bell, C. M. Shield, F. Busch, and G. Kruse, "A stochastic user equilibrium path flow estimator," *Transportation Research C*, vol. 5, no. 3–4, pp. 197–210, 1997.
- [24] M. G. H. Bell and Y. Iida, *Transportation Network Analysis*, John Wiley and Sons, Chichester, UK, 1997.
- [25] M. G. H. Bell and S. Grosso, "Estimating path flows from traffic counts," in *Proceedings of the International Workshop on Traffic and Mobility: Simulation-Economics-Environment*, pp. 85–105, Aachen, Germany, 1999.
- [26] D. I. Robertson, "TRANSYT: a traffic network study tool," RRL Report LR253, Transport and Road Research Laboratory, Crowthorne, UK, 1969.
- [27] R. A. Vincent, A. I. Mitchell, and D. I. Robertson, "User guide to TRANSYT version 8," TRRL Report LR888, Transport and Road Research Laboratory, Crowthorne, UK, 1980.
- [28] F. Teklu, A. Sumalee, and D. Watling, "A genetic algorithm approach for optimizing traffic control signals considering routing," *Computer-Aided Civil and Infrastructure Engineering*, vol. 22, no. 1, pp. 31–43, 2007.
- [29] H. Ceylan, "Developing combined genetic algorithm—Hill-climbing optimization method for area traffic control," *Journal of Transportation Engineering*, vol. 132, no. 8, pp. 663–671, 2006.
- [30] H. Ceylan and H. Ceylan, "A hybrid harmony search and TRANSYT hill climbing algorithm for signalized stochastic equilibrium transportation networks," *Transportation Research Part-C*, vol. 25, pp. 152–167, 2012.
- [31] R. Storn and K. Price, "Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.
- [32] T. J. Carrigan, B. H. Dennis, Z. X. Han, and B. P. Wang, "Aerodynamic shape optimization of a vertical-axis wind turbine using differential evolution," *ISRN Renewable Energy*, vol. 2012, Article ID 528418, 16 pages, 2012.
- [33] K. Guney and D. Karaboga, "New narrow aperture dimension expressions obtained by using a differential evolution algorithm for optimum gain pyramidal horns," *Journal of Electromagnetic Waves and Applications*, vol. 18, no. 3, pp. 321–339, 2004.
- [34] M. J. Reddy and D. N. Kumar, "Multiobjective differential evolution with application to reservoir system optimization," *Journal of Computing in Civil Engineering*, vol. 21, no. 2, pp. 136–146, 2007.
- [35] A. Ketabi and M. J. Navardi, "Optimization shape of variable capacitance micromotor using differential evolution algorithm," *Mathematical Problems in Engineering*, vol. 2010, Article ID 909240, 15 pages, 2010.
- [36] W. Gong, Z. Cai, and L. Zhu, "An efficient multiobjective differential evolution algorithm for engineering design," *Structural and Multidisciplinary Optimization*, vol. 38, no. 2, pp. 137–157, 2009.
- [37] L. Wang and L.-P. Li, "An effective differential evolution with level comparison for constrained engineering design," *Structural and Multidisciplinary Optimization*, vol. 41, no. 6, pp. 947–963, 2010.
- [38] J.-W. Wu, Y.-B. Zhang, and C. He, "Vehicle routing problem with time windows based on improved differential evolution algorithm," in *Proceedings of the 2nd International Conference on Advanced Computer Theory and Engineering (ICACTE '09)*, pp. 1311–1319, Cairo, Egypt, September 2009.
- [39] C. R. Suribabu, "Differential evolution algorithm for optimal design of water distribution networks," *Journal of Hydroinformatics*, vol. 12, no. 1, pp. 66–82, 2010.
- [40] A. Vasan and S. P. Simonovic, "Optimization of water distribution network design using differential evolution," *Journal of Water Resources Planning and Management*, vol. 136, no. 2, Article ID 015002QWR, pp. 279–287, 2010.
- [41] Ö. Kişi, "River suspended sediment concentration modeling using a neural differential evolution approach," *Journal of Hydrology*, vol. 389, no. 1–2, pp. 227–235, 2010.
- [42] C. Fisk, "Some developments in equilibrium traffic assignment," *Transportation Research B*, vol. 14, no. 3, pp. 243–255, 1980.
- [43] Y. Sheffi, *Urban Transportation Networks: Equilibrium Analysis with Mathematical Programming Methods*, Prentice-Hall, Upper Saddle River, NJ, USA, 1985.
- [44] F. V. Webster and B. M. Cobbe, "Traffic signals," Road Research Technical Paper no. 56, Ministry of Transport, London, UK, 1966.
- [45] D. G. Mayer, B. P. Kinghorn, and A. A. Archer, "Differential evolution—an easy and efficient evolutionary algorithm for model optimisation," *Agricultural Systems*, vol. 83, no. 3, pp. 315–328, 2005.

- [46] S. Das and P. N. Suganthan, "Differential evolution: a survey of the state-of-the-art" *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 1, pp. 4–31, 2011.

## Research Article

# Interesting Activities Discovery for Moving Objects Based on Collaborative Filtering

Guan Yuan, Shixiong Xia, and Yanmei Zhang

School of Computer Science and Technology, China University of Mining and Technology, No. 1, College Road, Xuzhou, Jiangsu 221116, China

Correspondence should be addressed to Shixiong Xia; [xiasx@cumt.edu.cn](mailto:xiasx@cumt.edu.cn)

Received 19 April 2013; Revised 16 June 2013; Accepted 6 July 2013

Academic Editor: Saeed Balochian

Copyright © 2013 Guan Yuan et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the development of location-based service, more and more moving objects can be traced, and a great deal of trajectory data can be collected. Finding and studying the interesting activities of moving objects from these data can help to learn their behavior very well. Therefore, a method of interesting activities discovery based on collaborative filtering is proposed in this paper. First, the interesting degree of the objects' activities is calculated comprehensively. Then, combined with the newly proposed hybrid collaborative filtering, similar objects can be computed and all kinds of interesting activities can be discovered. Finally, potential activities are recommended according to their similar objects. The experimental results show that the method is effective and efficient in finding objects' interesting activities.

## 1. Introduction

In the real world, with the development of location-based service, the activity of moving objects in some regions can be tracked and their motion trajectory will be recorded by positioning device. In a sense, the trajectory data of an object is its activities in the region. Therefore, the interesting activity discovery can be transferred to finding objects' interesting regions from their historical trajectories. Moving objects' activities are often along with some purposes. Therefore, finding and studying the interesting activities of moving objects can help to understand their behavior well [1, 2]. However, the moving object activities are of many characteristics. These characteristics can be obtained through analyzing the trajectory data. Therefore, it is very important to discover their interesting activities.

There are various kinds of activities for a moving object. However, different activity types are often along with different features, and these features cannot be analyzed from the traced trajectory directly. Therefore, finding interesting activities discovery for moving objects is a challenging task [3], which is mainly due to: (1) the sampled trajectory of moving object is expressed by  $\langle (Lat., Lon.), timestamp \rangle$ ; because of

its triviality, discovering moving objects' behaviors from their historical trajectories directly is very difficult [4]; (2) moving objects' interesting activities are not only known (which can be analyzed from the trajectory explicitly) but also unknown ones (some activities whose objects may be interesting but have not visited before), so it is difficult to find more complete interesting activities of objects synthetically [5].

In order to solve the problems mentioned previously, we propose a work on interesting activities discovery for moving objects based on collaborative filtering (CF, and in the rest of this paper we use CF to abbreviate collaborative filtering). First, we generate the objects' activity sequence and abstract the features of each activity on the basis of our previous work [3, 4]. Second, we build an object-activity-time matrix (OATM) with three dimensions, which are object identification, hot region, and sequence. Third, on the basis of the OATM, we compute object's interest degree to each hot region comprehensively and generate an object-region interest degree matrix (IDM). Fourth, combined with CF, similar objects can be queried and their common interesting activities can be found. Finally, a serial of potential interesting activities can be recommended to the moving objects according to the  $K$ -nearest neighbor (KNN) algorithm with

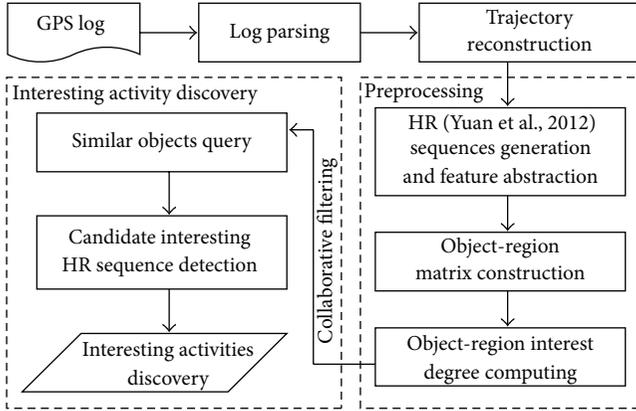


FIGURE 1: Framework of interesting activities discovery for moving objects.

a given threshold. As discussed above, the framework of the proposed work is described in Figure 1.

In summary, the contributions of this paper are as follows.

- (1) A moving object activity model is proposed, by which the redundant and trivial trajectories can be quantized and transformed to the meaningful moving object activity in a simple way.
- (2) The interesting drifting function is firstly introduced in the moving object interesting activity discovery, which can solve the problem of object interesting changes effectively.
- (3) The hybrid CF algorithm is introduced, taking full account on interesting drifting and interesting type, by which objects' interesting activity can be computed comprehensively and their potential activities can be discovered.

The rest of this paper is organized as follows. Section 2 describes the motivation of this paper and gives the related work of the topic. Section 3 presents an objects activities representation method and their interest degree computation method. Section 4 proposes the novel hybrid CF to the discovery of object's interesting activities. Section 5 conducts the experimental results on real data sets. Finally, Section 6 draws conclusions and points out some future works on moving object data mining.

## 2. The Motivation of the Work

The sampled trajectories are the record set of the moving objects, which can record their history activities with location, time, and other information. The related definitions on trajectory are given in [3, 4], which are also available in this paper. In Figure 2, there are two trajectories of Bob and Jim, and we can see that they share some common passed regions. Therefore, we can infer that the rest regions on their trajectories also have the possibility of mutual sharing by the two persons.

In current researches, there are many works that studied detecting the hot region from trajectory data [3, 5, 6]. Assuming that the regions are frequently visited by Bob and Jim, and

their geographical places are described in Figure 2. Therefore, we can make the following inferences: Bob generally eats breakfast at home, and then goes to the gym, then does shopping, and returns home after buying some food in a nearby supermarket. At the same time, Jim leaves home and goes to restaurant for his breakfast, and then does bodybuilding, next he goes shopping, in the early evening, he buys food in a nearby supermarket and goes home. These details show that bodybuilding and shopping are common hobbies shared by Bob and Jim. Therefore, these spatially overlapping regions indicate that the two persons have some similar interest activities; that is, they both love sports and shopping. If the two persons are similar enough in their interest activities, we can find one person's interest activities based on those of another person using the advanced hybrid CF [7–9].

With this idea, this paper presents a novel work on interesting activities discovery for moving objects based on hybrid CF.

## 3. Representation of Moving Object Activity

In our previous work, we have studied the moving pattern of moving objects from their trajectories [3, 4]; thus, we know that once an object stays in a hot region can be viewed as one of his activities. Therefore, we develop an efficient algorithm called DB-HR [3] to find the hot regions where moving objects stay in for a long time. In this section, we assume that objects' activities have already been generated by DB-HR through finding their hot regions. So, we need to formally represent activities for further analysis on the interesting activities finding for moving objects.

*Definition 1.* Activity ( $a$ ): an activity means a moving object's once access to a hot region, and it is a four-tuple, denoted as  $a = \{O_{id}, HR_{id}, StartTime, EndTime\}$ , where  $O_{id}$  is the identification of the object,  $HR_{id}$  is the hot region, and  $StartTime$  and  $EndTime$  are the timestamps that the object enters and leaves the hot region.

Using moving object activity, trajectory can be converted into a sequence of activities:  $S = \{a_1, a_2, a_3, a_4, \dots, a_n\}$ , where  $S$  is the activity sequence and  $a_i$  is the  $i$ th activity in the  $S$ . With the activity sequence, the triviality of trajectory can be avoided. Moreover, by calculating the activities in the same region of the activity sequence comprehensively, the quantitative calculation result is the interest degree of moving object for each region.

*3.1. Activity Matrix Generation.* In order to clearly express the spatiotemporal relationship between objects' activities, in this paper, we establish an object-activity-time matrix (OATM, as shown in Figure 3) to represent the objects' activities and times structurally. Then, we can calculate moving objects' interest degree in the activities by converting the object activity sequence into object activity matrix. In this way, on one hand, it is convenient for later calculation by converting the original multidimensional scattered information into a

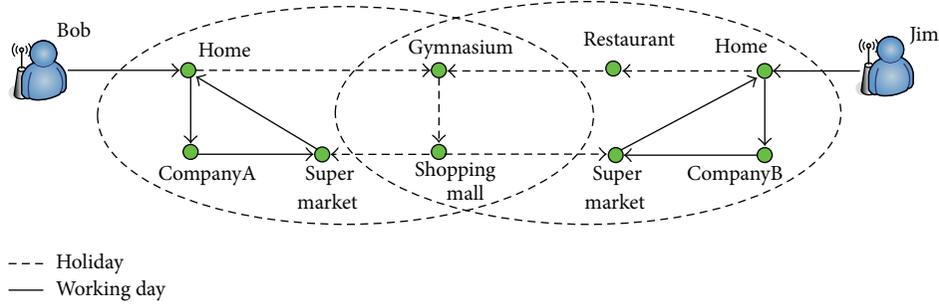


FIGURE 2: Relation between moving objects' activities.

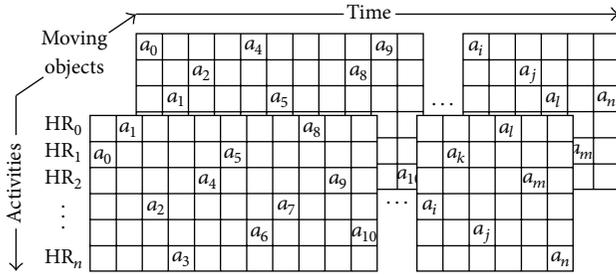


FIGURE 3: Matrix of activity hot regions and time frame for moving objects.

unified data; on the other hand, studying more comprehensive interest degree calculation method can intelligently quantify the interest degree of moving object activities, and the numerical value of interest degree can also reflect the characteristics of the object.

Figure 3 gives the matrix of activity regions and time frame for moving objects. It can be seen from the matrix that the moving object may access region  $HR_2$  many times in a given time period. By this way, we can know that the object may be more interested in this region. Therefore, we give the calculation rules of interest degree for object activity afterwards.

**3.2. Calculation of Interest Degree for Object Activities.** The more activities happened in a certain hot region, the higher interest degree it is for a moving object [5]. In this sight, we make further abstraction on activity types, then give the definition of object activity type (Activity Type, AT) and moving object activity interest degree (Interest Degree, ID):

**Definition 2 (activity type (AT)).** Activity type refers to the category of the hot regions, denoted as  $AT = \{at_i \mid HR_i, O_{id}, TimeLength, Count, LastTime\}$ . The attributes represent activity region, object identification, activity duration, number of activities, and the last access time, respectively. Activity type is mainly decided by the categories of hot region. For example, supermarket and shopping centers are both commercial areas, while persons in these two regions share different types of activities. Therefore, using activity type to classify the activities of moving objects can discover interesting activity of object from more general aspects.

**Definition 3 (interest degree (ID)).** The interest degree of moving objects in related regions is a measurement of how much the object likes the activity. The interest degree of moving object mainly includes three aspects: (1) the activity count that moving objects visited in a given hot region; (2) the activity duration that moving objects spent in the given hot region; (3) the activity time that moving objects spent in a given hot region.

The objects' interests are always changing and evolving constantly; in this paper, we introduce interest drift function [9] to solve this problem. On the basis of interest drift function, we give the calculation formula of interest degree for moving object. For example, the interest degree of object  $O$  in region  $HR_i$  is computed as follows:

$$ID_{HR_i}^O = \left( \frac{\sum_{\forall a.hr \in HR_i} C_{HR_i}}{\sum_{\forall a.o_{id} \in O} C_{at}^o} + \frac{\sum_{\forall a.hr \in HR_i} TL_{HR_i}}{\sum_{\forall a.o_{id} \in O} TL_{at}^o} \right) \times h(t). \quad (1)$$

In which,  $\sum C_{HR_i}$  is the total activities that the object  $O$  stayed in the hot region  $HR_i$ .  $\sum C_{at}^o$  represents object  $O$ 's total activities of in the data set.  $\sum TL_{HR_i}$  denotes the total duration of the object  $O$  in the  $HR_i$ .  $\sum TL_{at}^o$  represents the total duration of the object  $O$  in the data set.  $h(t)$  is the interesting drift function, and it is the function of time  $t$ . Through the function, the retention degree of the object activity in the interesting region  $HR_i$  can be calculated given the object interesting drift function  $h(t)$  as follows:

$$h(t) = m \times \left( \frac{t - t_{\min}}{t_{\max} - t_{\min}} \right)^2 + 1 - m. \quad (2)$$

In (2),  $t_{\min}$  is the earliest time that the moving object access to related hot region, and  $t_{\max}$  is the most recent statistical time. Let  $DS$  be the set of statistical data, then  $t_{\min} = DS.BeginTime$ , and  $t_{\max} = DS.EndTime$  ( $t_{\min} \leq t \leq t_{\max}$ ).  $m$  is the coefficient of interests drifting, namely, the speed of object's interest drift,  $m \in [0, 1]$ . When  $m = 0$ , no interests drift happens, and when  $m = 1$ , the fully nonlinear interest drift happens; the function value of interest drift is between 0 and 1. According to the definition of interest drift coefficient [10], each object's interest drift coefficient is not the same. In this regard, this paper uses parameter  $m$  to denote different interest drift.

In order to make it more flexible to calculate the object's interest degree to a certain activity, we need to introduce

two factors to adjust the importance of visit count and visit time. Therefore, we define the weighting vector of the interest degree,  $W = \{W_C, W_{TL}\}$ .  $W_C$  is the weight of the visit count, and  $W_{TL}$  is the weight of visit time. The interest degree of object activity can be adjusted by setting different numerical ratios. Therefore, we give the interest degree formula as follows:

$$\begin{aligned} ID_{HR_i}^O = & \left( W_C \times \frac{\sum_{\forall a, hr \in HR_i} C_{HR_i}}{\sum_{\forall a, o_{id} \in O} C_{at}^o} \right. \\ & \left. + W_{TL} \times \frac{\sum_{\forall a, hr \in HR_i} TL_{HR_i}}{\sum_{\forall a, o_{id} \in O} TL_{at}^o} \right) \times h(t). \end{aligned} \quad (3)$$

By setting the weighting vector of the interest degree, the interest degree of an object in a particular region can be calculated by a more comprehensive and more flexible way. By using (3), the matrix in Figure 3 can be converted into matrix of -hot regions objects, shown in Figure 4. By this way, the activities of objects in their interesting regions are converted from multidimensional sampled data into a single numerical data, and it is convenient for further calculation.

In practical application, due to the randomness of object activity and the longer time interval, the matrixes in Figures 3 and 4 are often very sparse. In fact, the object activities are all of specific purposes; for example, the objects no matter how long they stay in a supermarket or in a shopping center, they all do shopping. Therefore, if we give full consideration into the type of activity, it may increase the accuracy of the interesting activities discovery, and at the same time, it can reduce a great deal of redundancy. Suppose that  $HR_1$ ,  $HR_2$ ,  $HR_3$ , and  $HR_4$  which are interesting regions of  $O_1$ ,  $HR_1$ , and  $HR_2$  belong to the same type  $T_1$ , while  $HR_2$  and  $HR_3$  belong to the same type  $T_2$ , then the moving object in the two kinds of activity types has a comprehensive interest degree, expressed as follows:

$$\overline{ID}_{T_m}^O = \frac{\sum_{HR_i \in T_m} (\sum_{\forall a, hr \in HR_i} C_{HR_i} \times ID_{HR_i}^O)}{\sum_{HR_i \in T_m} \sum_{\forall a, hr \in HR_i} C_{HR_i}}. \quad (4)$$

The object's total interest degree to a certain activity type is the average weighted values of its interest degree to each single activity, which belongs to the same activity type. This value can synthetically reflect the comprehensive level of interest degree that the moving objects visit to a certain type of regions. The interest degree of activity type-objects can be calculated by (4).

Through the comprehensive calculation for interest degree, the interest degree matrix of hot regions-objects (Figure 4) and the interest degree matrix of activity type (Figure 5) are generated. Afterwards we will use hybrid CF to calculate the similarity degree of moving object based on their interesting matrixes and find their interesting activities.

#### 4. Interesting Activity Discovery for Moving Objects

To better describe the work proposed in this paper, notation of basic symbols is given in Table 1.

TABLE 1: Parameters notation.

Parameter	Description
$S_o$	The activity sequence of moving objects
$Len(S)$	The length of the activities sequence
$T(S)$	Time span of activity sequence
HRset	The set of hot regions
$S_i^o$	The $i$ th activity of object $O$ in the sequence $S$
$Len(HRset)$	The total count of activities in the HRset

Moving objects  $\longrightarrow$

	$O_0$	$O_1$	$O_2$	$\dots$	$O_i$	$\dots$	$O_m$
$HR_0$	$ID_0^0$	$ID_0^1$	$ID_0^2$	$\dots$	$ID_0^i$	$\dots$	$ID_0^m$
$HR_1$	$ID_1^0$	$ID_1^1$	$ID_1^2$	$\dots$	$ID_1^i$	$\dots$	$ID_1^m$
$HR_2$	$ID_2^0$	$ID_2^1$	$ID_2^2$	$\dots$	$ID_2^i$	$\dots$	$ID_2^m$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\dots$	$\vdots$	$\dots$	$\vdots$
$HR_n$	$ID_n^0$	$ID_n^1$	$ID_n^2$	$\dots$	$ID_n^i$	$\dots$	$ID_n^m$

Activities  $\downarrow$

FIGURE 4: The interest degree matrix of hot regions-objects.

**4.1. Interesting Activities Discovery.** In this section, we use hybrid CF method to find the potential interesting activities for moving objects. Similarity computation is the key of collaborative filtering method. The cosine similarity is often viewed as the standard similarity function for collaborative filtering algorithm [7, 11], and it uses the angle between two vectors to represent their similarity. The function is shown as follows:

$$\text{Sim}(O_a, O_b) = \frac{\sum_{i \in I} ID_i^a \times ID_i^b}{\sqrt{\sum_{i \in I} (ID_i^a)^2} \sqrt{\sum_{i \in I} (ID_i^b)^2}}. \quad (5)$$

Formula (5) is the standard cosine similarity function.  $ID_i^a$  represents the interest degree of moving objects  $O_a$  in hot region  $HR_i$ . In order to avoid the differences in moving objects while visiting different hot regions, we need to improve the cosine similarity function. Pearson [11] proposed the modified cosine similarity function on the basis of the standard one as follows:

$$\text{Sim}(O_a, O_b) = \frac{\sum_{i \in I} ((ID_i^a - \overline{ID}_a) \times (ID_i^b - \overline{ID}_b))}{\sqrt{\sum_{i \in I} (ID_i^a - \overline{ID}_a)^2} \times \sqrt{\sum_{i \in I} (ID_i^b - \overline{ID}_b)^2}}. \quad (6)$$

Among them,  $\overline{ID}_a$  and  $\overline{ID}_b$  are the average interest degrees of moving objects  $O_a$  and  $O_b$  in all hot regions. The adjusted cosine similarity function improves the interest degree deviation of different objects by introducing the average interest degree. We use  $O_a$  and  $O_b$  commonly visited regions as their interest activities ( $HR_a \cap HR_b$ ), denoted as  $HR'$ . The value of  $\text{Sim}(O_a, O_b) \in [0, 1]$ , and the bigger  $\text{Sim}(O_a, O_b)$  is, the higher similarity between  $O_a$  and  $O_b$  is. Through the similar function

		Moving objects						
		$O_0$	$O_1$	$O_2$	$\dots$	$O_i$	$\dots$	$O_m$
Activity type	$T_0$	$\overline{ID}_0^0$	$\overline{ID}_0^1$	$\overline{ID}_0^2$	$\dots$	$\overline{ID}_0^i$	$\dots$	$\overline{ID}_0^m$
	$T_1$	$\overline{ID}_1^0$	$\overline{ID}_1^1$	$\overline{ID}_1^2$	$\dots$	$\overline{ID}_1^i$	$\dots$	$\overline{ID}_1^m$
	$T_2$	$\overline{ID}_2^0$	$\overline{ID}_2^1$	$\overline{ID}_2^2$	$\dots$	$\overline{ID}_2^i$	$\dots$	$\overline{ID}_2^m$
	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\dots$	$\vdots$	$\dots$	$\vdots$
	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\dots$	$\vdots$	$\dots$	$\vdots$
	$T_n$	$\overline{ID}_n^0$	$\overline{ID}_n^1$	$\overline{ID}_n^2$	$\dots$	$\overline{ID}_n^i$	$\dots$	$\overline{ID}_n^m$

FIGURE 5: The interest degree matrix of activity type-objects.

		Moving objects						
		$O_0$	$O_1$	$O_2$	$\dots$	$O_i$	$\dots$	$O_m$
Moving objects	$O_0$	1						
	$O_1$	$S_{10}$	1					
	$O_2$	$S_{20}$	$S_{21}$	1				
	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\dots$	1		
	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\dots$	$\vdots$	$\dots$	1
	$O_m$	$S_{m0}$	$S_{m1}$	$S_{m2}$	$\dots$	$S_{mi}$	$\dots$	$\dots$

FIGURE 6: Similarity matrix of moving objects.

Formula (6), we can get an object similarity matrix (shown as Figure 6), which contains object similarity between each other. For the similarity is symmetrical, the similarity matrix can be showed as a low triangular matrix.

There are two major tasks in finding object interesting regions: (1)  $\forall HR, \exists HR_i \rightarrow O_a$  has never accessed ( $HR_i \notin S_{O_a, HRset}$ ); then we will predict the interest degree of object  $O_a$  in  $HR_i$ ; (2) finding  $N$  ( $N > 1$ ) regions that object  $O_a$  most likely to visit from the hot regions which it never visited, as the potential interest regions of object. The similar matrix is symmetrical, and we can easily determine the most similar objects. Here, we use  $K$  to denote the count of similar objects.

By calculating the similarity between moving objects, the selection of  $k$  similar moving objects for unknown hot region  $HR_i$  based on hybrid CF is proposed (the  $K$  moving objects which are most similar to  $O_a$ ). We define these similar objects as  $S(O_a)$  and  $|S(O_a)| \leq k$ ; then we can forecast their interest degree in  $HR_i$ , expressed by  $\overline{ID}_j^a$ , and is shown as follows:

$$\overline{ID}_j^a = \overline{ID}_a + \frac{\sum_{O_x \in S(O_a)} \text{sim}(O_a, O_x) \times (\overline{ID}_j^x - \overline{ID}_x)}{\sum_{O_x \in S(O_a)} \text{sim}(O_a, O_x)}. \quad (7)$$

In (7),  $\overline{ID}_a$  and  $\overline{ID}_x$  are the average interest degrees of  $O_a$  and  $O_x$  to other hot regions;  $\overline{ID}_x^a$  is the predictive interest degree of  $O_a$  to  $HR_x$ . In the process of interesting activity discovery, we generally select  $k$  most similar neighbors in the similarity matrix, use the  $k$  nearest neighbors to recommend predictive interesting activity, and give the predictive interest degree.

4.2. *Flow of the Proposed Algorithm.* Therefore, based on the above analysis, we give the approach of interesting activity discovery for moving objects based on hybrid CF, shortly for approach of interesting activity discovery (IAD). Through IAD algorithm, potential interesting activities can be discovered for objects. The pseudocode of IAD algorithm is shown in Algorithm 1.

Algorithm IAD requires 3 input parameters, which are the recommended object  $O_{id}$ , the number of the nearest neighbor  $k$ , and the threshold of predictive interest degree  $ID_{th}$ . This algorithm mainly includes the following steps. Lines 01–04 are mainly used to load sequence of moving object activity and extract related activities; then calculate the interest degree of object-activity and the similarity between objects. The calculated data is stored in the matrix, which is convenient for further analysis. Lines 05–07 are mainly used to predict the potential interesting activities based on the object similarity matrix and the object interest degree matrix. Also, the predictive interest degree for potential activities is given. Since the algorithm only calculates the activities' interest degree for moving objects and discoveries of objects' interesting activities, therefore, we just need to update the calculated interest degree. The worst case of the time complexity is  $O(m^2 + n^2)$ , where  $m$  is the number of moving object and  $n$  is the total of hot regions. After the first calculation, the time complexity of the algorithm is a constant  $O(m+n)$ , where  $m$  is the time consumption of finding similar neighbor and  $n$  is the time of recommending the potential interesting activities of objects.

## 5. Experiments and Analysis

To validate the method proposed in this paper, we develop a module of interesting activities discovery for moving objects in our trajectory data mining system (TrajMiner). The data set is the GPS trajectory data [12], abbr. GeoLife. The metadata of the trajectory is stored in the text files. TrajMiner firstly transformed the GeoLife metadata into the standard trajectory data by noise reduction, data cleaning, and trajectory reconstruction. Then, it stores the trajectory data into the SQLServer 2008. In this experiment, we select 2983 trajectories of 50 moving objects, which include 6037452 sampling points, to analyze the accuracy and efficiency of the proposed method.

5.1. *Accuracy Analysis.* In this part, we adopt precision and recall functions to validate the accuracy of IAD. The precision and recall functions are given as follows:

$$\text{Precision} = \frac{\sum_i TP_i}{\sum_i (TP_i + FP_i)} \times 100\%, \quad (8)$$

$$\text{Recall} = \frac{\sum_i FN_i}{\sum_i (TP_i + FN_i)} \times 100\%,$$

where  $TP_i$  denotes true positive, which means that the object has visited  $HR_i$  and the algorithm was truly recommended to him.  $FP_i$  denotes false positive, which means that the object did not visit  $HR_i$ , but it was truly recommended to him.  $FN_i$  is

**Input:**  $O_{id}$ ,  $k$ ,  $ID_{th}$   
**Output:** the latent activity set of  $O_{id}$   
01: Abstract all the activity sequences and stored them in the  $S_{set}$ ;  
02: Build object-visit-matrix  $M$  according to the  $S_{set}$ ;  
03: Compute object-region interest degree from  $M$ , and build the interest degree matrix  $M_{id}$ ;  
04: Search similar objects using hybrid CF algorithm, and generate similar matrix  $M_s$  according to  $M_{id}$ ;  
05: Find the  $k$  most nearest neighbors to  $O_{id}$  from  $M_s$ ;  
06: On the basis of the  $k$  most nearest neighbors, recommend the latent activities for  $O_{id}$ ;  
07: If the  $O_{id}$ 's interesting degree to the recommended activities is less than  $ID_{th}$ , then remove the recommended activity, otherwise, recommend them to  $O_{id}$ .  
End.

ALGORITHM 1: Interesting activity discovery (IAD).

TABLE 2: Data volume of the six stages.

Phase	Count of objects	Count of trajectories	Count of sampling points
1		1080	1,000,170
2		2091	2,004,283
3	50	2319	3,002,400
4		2672	4,004,758
5		2870	5,001,380
6		2983	6,037,452

false negative, which means that the object have visited  $HR_i$ , while it was not recommended to him. Therefore, the higher precision is, the better our method is, and the lower recall is, the better our method is. Figure 7 shows the accuracy of our proposed method.

In Figure 7,  $P$  and  $R$  are the precision and recall of the  $ID_{th}$  value noted in the corresponding bracket. We analyze the accuracy in two situations, one is associated with interesting drift (Figure 7(a)), and the other is without interesting drift (Figure 7(b)). From the two figures, we can see that with the increase of  $K$  and  $ID_{th}$ , the precision increases slightly, and the recall decreases slowly. The precision associated with interesting drift seems smooth (with  $m = 0.5$ ), while the precision without interesting drift is not very stable.

**5.2. Efficiency Analysis.** In order to validate the efficiency of the proposed algorithm, in this section, we adopt the incremental validation method for IAD verification. First, we select a subset of GeoLife for static calculation of interesting activity. The subset includes 1000170 sampling points from 1080 tracks of 50 objects, and 104 hot regions can be discovered by the DB-HR [3] from trajectories. Then, we increase the number of training data on the basis of the first phase gradually from 1000 to nearly 3000 trajectories of the 6 stages. For each training stage, we analyze the efficiency of the algorithm. Table 2 gives the number of training data in different stages.

In order to evaluate the effectiveness of IAD algorithm, we introduce mean absolute error (MAE) [11, 13] as the efficiency measurement. MAE is the most commonly used

efficiency measurement of recommendation because it is easy to understand and it can be intuitive, convenient, and accurate to evaluate the quality of algorithm. MAE is the interest degree difference between the predictive activities and real activities. Therefore, the smaller MAE is the higher efficiency the algorithm is. In order to compare the experimental results, this section improves (4) by removing the interest shift function  $h(t)$ . Getting the simple interest degree of the object  $\vec{ID}_{HR_i}^O$ , both of the weights are set to 0.5:

$$\vec{ID}_{HR_i}^O = 0.5 \times \frac{\sum_{\forall a, hr \in HR_i} C_{HR_i}}{\sum_{\forall a, o_{id} \in O} C_{at}^o} + 0.5 \times \frac{\sum_{\forall a, hr \in HR_i} TL_{HR_i}}{\sum_{\forall a, o_{id} \in O} TL_{at}^o}. \quad (9)$$

We set the predictive interest degree of activities for object as  $\{P_1, P_2, \dots, P_n\}$ . While the actual interest degree set between object activity is  $\{R_1, R_2, \dots, R_n\}$ , then according to the definition of MAE, the mean absolute deviation MAE is calculated as follows:

$$MAE = \frac{\sum_{i=1}^N |P_i - R_i|}{N}. \quad (10)$$

In the experiment, we set  $k \in [3, 10]$  and  $ID_{th} \in [0.5, 1]$  to guide the interesting activities discovery. First, we make a comment on the average similarity of each stage under the same value of  $k$ . Then, we compare the impact of different values of  $k$  on MAE, trying to find out the relationship between the number of the nearest neighbors and the efficiency of recommendation, and the effect of similarity on MEA under different values of  $k$ . The contrast relation between  $k$  and MAE in different incremental stages is shown in Figure 8. As being the distinguished degree of results when  $k = 3, 5, 8$ , in Figure 8(b), the predicted MAE is relatively high. As shown in Figure 8, when  $k = 8$ , the average of similarity between objects compared to the stability number is changed relatively little. It shows that in this case the similarity between objects is more consistent, and increased activity is similar in the neighborhood with the increase in the number. Therefore, it will achieve good effect to recommend among interesting activities for object using the number of neighbor with  $k = 8$ . For Figure 8(b), the recommendation of interesting activities depends on the size

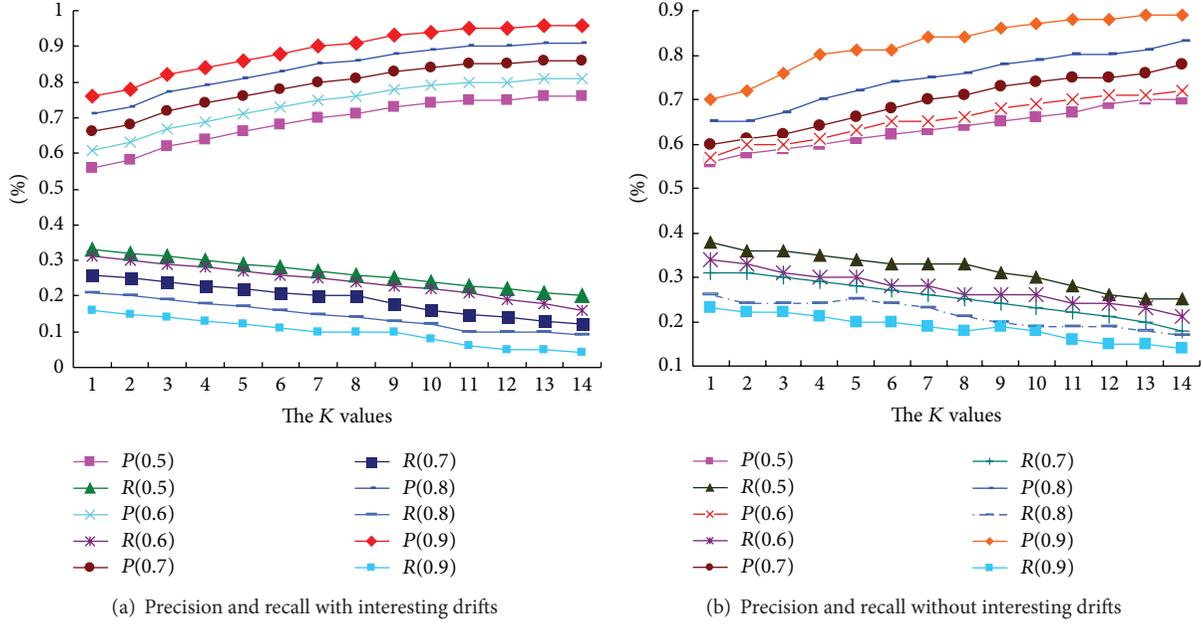


FIGURE 7: Accuracy analysis of the proposed method.

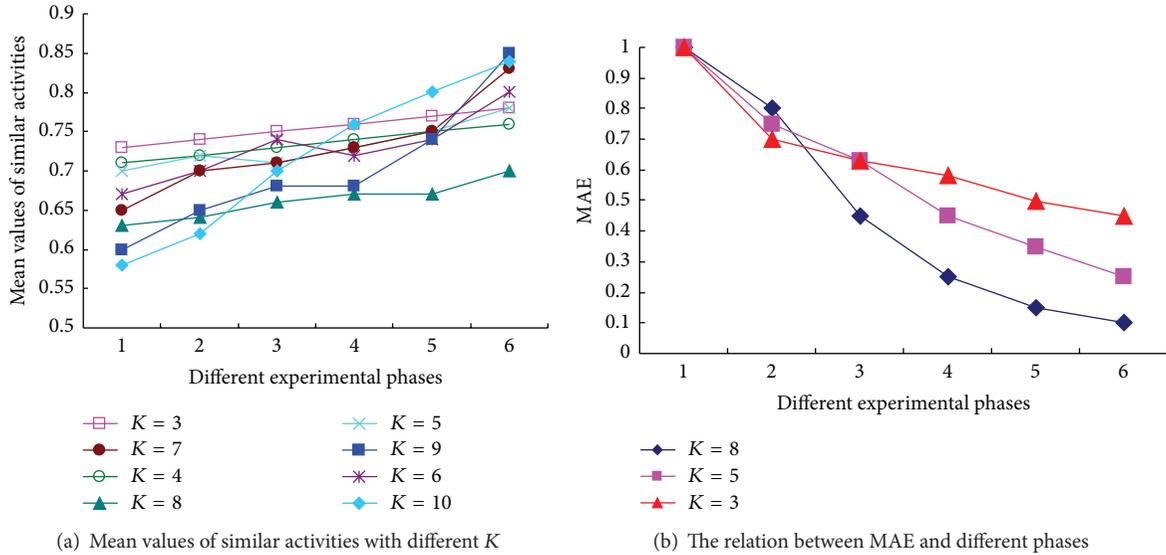


FIGURE 8: The MAE values with different phases and K.

of training data set; the larger the number of the training data is, the better effect is and the more accurate results are. As can be seen from Figure 8, with the increase of data, the accuracy of interesting activities discovery tends to be stable. In this case, it can achieve more ideal effect with data continuing to increase.

**5.3. Effectiveness Analysis.** In order to validate the effectiveness of the proposed algorithm, in this section, we compare our algorithm with traditional CFs and KNN-CF [11] (which is a recent proposed advanced CF). Here, we denote UBCF (user-based CF) and IBCF (item-based CF) as traditional

CF algorithms. In this section, another evaluation index coverage [13] is introduced to validate the effectiveness of the compared algorithms. Coverage is a widely used effectiveness evaluation index to check the coverage rate of a recommendation system, the calculation is shown as follows:

$$\text{Coverage} = \frac{\sum_{o_{id} \in O} |P_{o_{id}} \cap I_{o_{id}}|}{\sum_{o_{id} \in O} I_{o_{id}}}. \quad (11)$$

In the formula above,  $P_{o_{id}}$  is the activity set that the system recommends to the  $O_{id}$ , and  $I_{o_{id}}$  is the interesting activity set discovered by the system.  $P_{o_{id}} \cap I_{o_{id}}$  is the coverage of

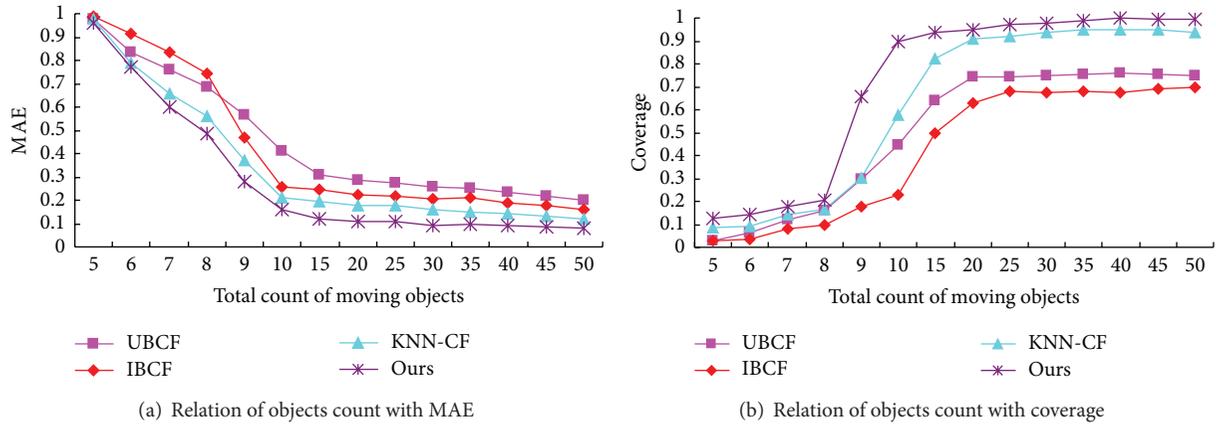


FIGURE 9: Effectiveness analysis with two indexes.

the system recommends and objects' interesting activities. Therefore, the higher coverage is, the more effective algorithm is.

In this experiment, we validate the effectiveness of our hybrid CF from two aspects. (1) the convergence of the compared algorithm with the relation of MAE; (2) the effectiveness of the proposed algorithm with the relation of coverage.

We can read from Figure 9 that our algorithm is better than traditional CFs and recent proposed advanced KNN-CF. In Figure 9(a), the 4 algorithms perform nearly at the same level. When the total count of moving objects is 10 or more, the MAEs of hybrid CF and KNN-CF decrease rapidly and the two algorithms become smooth when total objects are more than 30. Since no strategies are used in traditional CFs, both UBCF and IBCF perform not so well as ours and KNN-CF. The best MAE value of ours is 0.09 when total objects reach 30, while the best MAE value of KNN-CF is 0.13 when total objects reach 37. In Figure 9(b), we can see that our hybrid CF reaching the stable state is faster than the other 3 algorithms. In both experiments of Figure 9, the difference between 4 algorithms is very little when total objects are very few. As shown in Figure 9(b), the best coverage of our hybrid CF reaches 99.8% when the count of objects is more than 40, and the best coverage of KNN-CF reaches 91.3% when the count of objects is more than 35. Since activity type and interesting drifting mechanism is adopted in our algorithm, interesting hot regions which objects visit frequently will have more weights, and this value plays very important role in the object interesting activity discovery. Therefore, our hybrid CF is better than KNN-CF and traditional CFs, such as UBCF and IBCF.

As discussed in [14, 15], the performance of CF algorithm is often influenced by three factors: (1) the data sparseness; (2) the scalability of the work; (3) the original evaluation. In fact, the mentioned three factors are well conquered in our work. Firstly, we use activity type to merge the sparse object's interesting into once class or type. Secondly, through the activity type, a great many activities happened in hot regions with the same type can also be merged, and so, the performance can be well improved. Thirdly, for a hot region

never visited by anyone, it also can be inferred to the certain type and recommended to objects who like it.

## 6. Conclusions

In this research, a novel work on interesting activity discovery for moving object is presented. Firstly, a moving object activity model is proposed to store objects' activities and their attributes. Secondly, by introducing activity type and interesting drifting mechanism, the hybrid CF is proposed to compute objects interesting activities comprehensively. Thirdly, based on newly proposed hybrid CF, objects which are similar to each other can be found, and their potential interesting regions can be predicted. We have conducted numerous experiments from several aspects on real dataset and demonstrated the accuracy, efficiency, and effectiveness of our hybrid CF. In the future study, we can address moving objects' interesting routes discovery based on the work proposed in this paper.

## Acknowledgment

This work was supported by the Fundamental Research Funds for the Central Universities, China (under Grant no. 2013QNA25).

## References

- [1] Z. H. Li, B. L. Ding, J. W. Han, R. Kays, and P. Nye, "Mining periodic behaviors for moving objects," in *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '10)*, pp. 1099–1108, New York, NY, USA, July 2010.
- [2] H. P. Cao, N. Mamoulis, and D. W. Cheung, "Discovery of periodic patterns in spatiotemporal sequences," *IEEE Transactions on Knowledge and Data Engineering*, vol. 19, no. 4, pp. 453–467, 2007.
- [3] G. Yuan, S. X. Xia, L. Zhang, M. Zhu, and C. Ji, "Asynchronous periodic patterns discovery for moving objects," *Journal of Convergence Information Technology*, vol. 7, no. 9, pp. 286–294, 2012.

- [4] G. Yuan, S. X. Xia, L. Zhang, Y. Zhou, and C. Ji, "An efficient trajectory-clustering algorithm based on index tree," *Transactions of the Institute of Measurement and Control*, vol. 32, no. 7, pp. 850–861, 2012.
- [5] Y. Zheng and X. Xie, "Learning location correlation from GPS trajectories," in *Proceedings of the 11th IEEE International Conference on Mobile Data Management (MDM '10)*, pp. 27–32, Kansas, Mo, USA, May 2010.
- [6] A. T. Palma, V. Bogorny, B. Kuijpers, and L. O. Alvares, "A clustering-based approach for discovering interesting places in trajectories," in *Proceedings of the 23rd Annual ACM Symposium on Applied Computing (SAC '08)*, pp. 863–868, Fortaleza, Brazil, March 2008.
- [7] C.-G. Huang, J. Yin, J. Wang, Y.-B. Liu, and J.-H. Wang, "Uncertain neighbors' collaborative filtering recommendation algorithm," *Jisuanji Xuebao/Chinese Journal of Computers*, vol. 33, no. 8, pp. 1369–1377, 2010.
- [8] H. Wu, Y.-J. Wang, Z. Wang, X.-L. Wang, and S.-Z. Du, "Two-phase collaborative filtering algorithm based on co-clustering," *Ruan Jian Xue Bao/Journal of Software*, vol. 21, no. 5, pp. 1042–1054, 2010.
- [9] Q. N. Li, Y. Zheng, X. Xie, Y. K. Chen, W. Y. Liu, and Y. Ma, "Mining user similarity based on location history," in *Proceedings of the 16th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (ACM GIS '08)*, pp. 34–43, Irvine, Calif, USA, November 2008.
- [10] G. Yuan, S. X. Xia, L. Zhang, Y. E. Li, and J. Y. Xiao, "Web users clustering based on interest degree," *Microelectronics & Computer*, vol. 25, no. 9, pp. 112–114, 2008.
- [11] X. Luo, Y.-X. Ouyang, Z. Xiong, and M. Yuan, "The effect of similarity support in k-nearest-neighborhood based collaborative filtering," *Jisuanji Xuebao/Chinese Journal of Computers*, vol. 33, no. 8, pp. 1437–1445, 2010.
- [12] S. Karli and Y. Saygin, "Mining periodic patterns in spatio-temporal sequences at different time granularities," *Intelligent Data Analysis*, vol. 13, no. 2, pp. 301–335, 2009.
- [13] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl, "Evaluating collaborative filtering recommender systems," *ACM Transactions on Information Systems*, vol. 22, no. 1, pp. 5–53, 2004.
- [14] H.-L. Xu, X. Wu, X.-D. Li, and B.-P. Yan, "Comparison study of internet recommendation system," *Ruan Jian Xue Bao/Journal of Software*, vol. 20, no. 2, pp. 350–362, 2009.
- [15] X. Y. Su and T. M. Khoshgoftaar, "A survey of collaborative filtering techniques," *Advances in Artificial Intelligence*, vol. 2009, Article ID 421425, 19 pages, 2009.

## Research Article

# Surface Defect Target Identification on Copper Strip Based on Adaptive Genetic Algorithm and Feature Saliency

Xuewu Zhang, Wei Li, Ji Xi, Zhuo Zhang, and Xinnan Fan

*Computer and Information College, Hohai University, Changzhou 213022, China*

Correspondence should be addressed to Xuewu Zhang; [lab\\_112@126.com](mailto:lab_112@126.com)

Received 22 February 2013; Accepted 21 June 2013

Academic Editor: Yudong Zhang

Copyright © 2013 Xuewu Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

To enhance the stability and robustness of visual inspection system (VIS), a new surface defect target identification method for copper strip based on adaptive genetic algorithm (AGA) and feature saliency is proposed. First, the study uses gray level cooccurrence matrix (GLCM) and HU invariant moments for feature extraction. Then, adaptive genetic algorithm, which is used for feature selection, is evaluated and discussed. In AGA, total error rates and false alarm rates are integrated to calculate the fitness value, and the probability of crossover and mutation is adjusted dynamically according to the fitness value. At last, the selected features are optimized in accordance with feature saliency and are inputted into a support vector machine (SVM). Furthermore, for comparison, we conduct experiments using the selected optimal feature subsequence (OFS) and the total feature sequence (TFS) separately. The experimental results demonstrate that the proposed method can guarantee the correct rates of classification and can lower the false alarm rates.

## 1. Introduction

With the development of production and processing, the quality requirements of products appearance have become increasingly higher, which caused the manufacturers to ensure nondestructive inspection for providing products. However, due to the variety of conditions in VIS, the size and the shape of the defect targets in detected object are usually indefinite, especially the false targets. For example, the falling iron scraps, the flying moth, the oil droplet, and so forth in copper strip production lines might be classified as true targets. Therefore, an automatic target identification system is necessary to be designed to eliminate the false targets. In this paper, the target identification system has the following stages: feature extraction, feature selection, and feature optimization and classification.

Feature extraction is a key factor for defect inspection. Actually, metal surface defects are mostly similar to texture patterns. Numerous methods have been proposed to extract textural features, and cooccurrence matrix method is one of statistical methods. Huang et al. [1] proposed an inspecting technology using gray level cooccurrence matrix to extract colorific and structural textures for solders of a flexible

printed circuit (FPC). The experiments indicated that the defective solder is obviously different from the nondefective solder in several kinds of quantified characters. Zhang et al. [2] developed a vision inspection system for the surface defects of strongly reflected metal. In this system, spectral measure approach based on Fourier spectral is used to compute textures which are inputted into a SVM to detect defects. Moment is a linear characteristic which has translation invariance, scaling invariance, and rotation invariance. Ping et al. [3] used the moment invariants to pick up the characters of typical copper surface defects. Tolba et al. [4] used GLCM and HU invariant moments based on Learning Vector Quantization (LVQ) classifiers to detect defects of textiles, and the correct defect detection rate is 98.64% with an average false acceptance rate of 0.0012.

Feature selection is an important preprocessing step in target identification system (TDS). Selecting an OFS that preserves classification accuracy is a growing important problem because of the increasing size and dimensionality of real data sets. Achieving reduction of the relevant features number without negative effect on classification accuracy is a goal which greatly improves the overall effectiveness of the target identification system. After analyzing the advantages

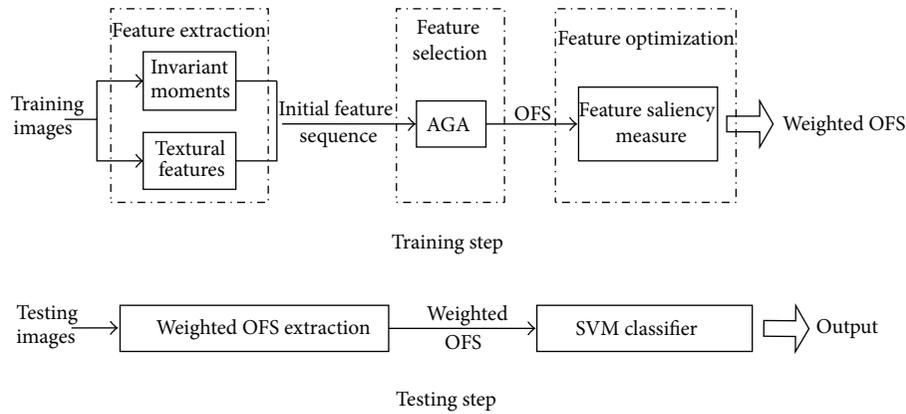


FIGURE 1: Block diagram of target identification.

and disadvantages of filtering and wrapping approaches for feature selection, Wang and Zheng proposed a hybrid approach named filtering-wrapping feature selection (FWFS) [5]. This approach uses information gain to evaluate feature's relevance to the target class and mutual information to evaluate redundancies among features. Then, the actual classifier is used as a "black box" to evaluate the fitness value. Rodriguez et al. [6] proposed a feature selection method, named Quadratic Programming Feature Selection (QPFS), which can limit the computational complexity for large data sets. Comparatively speaking, genetic algorithm has strong ability in local search, and the local optimal solution can be avoided in searching process [7], it solves the shortcomings of slow convergence speed and high time complexity. Yang et al. [8] presented an improved genetic algorithm (IGA) to select the optimal feature subset effectively and efficiently from a multicharacter feature set (MCFS). The IGA adopts segmented chromosome management scheme to implement local management of chromosome. A segmented crossover operator and a segmented mutation operator are employed to operate on these segments to avoid invalid chromosomes. The feature selection step of Zhang's method is based on adaptive simulated annealing genetic algorithm, which can guarantee the correct rate of classification and can improve the efficiency [9].

The proposed method optimizes the OFS judging by the feature saliency (i.e., the contribution degree of feature), which can increase the accuracy rate and robustness of the TDS. Saliency is the human perception of certain quality and quantity which can representatively reflect the difference between targets and other ones. For performance comparison, we do experiments on defects detection using the OFS and the TFS.

The organization of this paper is as follows: in Section 2, the proposed method is explained by three steps, feature extraction, feature selection, and feature optimization; Section 3 shows the experimental results and discusses the performance of the proposed method; conclusions are given in Section 4.

## 2. Target Identification System Design

Figure 1 shows the target identification process that contains two steps, training step and testing step. In the training step, the identification system firstly extracts the initial features from each image in training set to form the initial feature sequence set, which is then inputted into feature selection to obtain the optimal feature sequence and the weights. In the testing step, the weighted OFS is inputted into SVM classifier to implement the target identification.

- (1) *Feature extraction.* This study uses invariant moments and textural features to ensure that the extracted feature sequence can satisfy the requirements of target identification with validity, less computation quantity, and good robustness.
- (2) *Feature selection.* Genetic algorithm is a faster global optimizing algorithm with constant feedback correction. Consequently, the paper adopts GA to select the optimal feature sequence.
- (3) *Feature optimization.* Generally, each feature in OFS has different contribution to target identification. Therefore, this paper measures the feature saliency as the weights to update the existing identification model, which can guarantee the robustness and can further improve the performance of the identification model.
- (4) *Classification and identification.* SVM has been proposed as popular tools for classification problems [10]. After eliminating the features with less contribution in the training step, SVM can identify the target more accurately.

## 3. Methodology

### 3.1. Feature Extraction

**3.1.1. HU Invariant Moments.** Moments can be used for rep-representation of a two-dimensional image on the basis of

the Papoulis uniqueness theorem [11]. Invariant moment theory based on region shape recognition was proposed by Hu [12] at first.

The  $(p + q)$ th-order central moment of digital images is defined as

$$\mu_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q \rho(x, y). \quad (1)$$

The normalized central moment is

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}}, \quad p, q = 1, 2, 3, \dots \quad (2)$$

Hu proposed seven invariant moments which satisfied the conditions of translation invariance, scaling invariance, and rotation invariance. These moments can be written as follows:

$$\begin{aligned} \phi_1 &= \mu_{20} + \mu_{02}, \\ \phi_2 &= (\mu_{20} + \mu_{02})^2 + 4\mu_{11}^2, \\ \phi_3 &= (\mu_{30} - 3\mu_{11})^2 + (3\mu_{21} - \mu_{03})^2, \\ \phi_4 &= (\mu_{30} + \mu_{12})^2 + (\mu_{21} + \mu_{03})^2, \\ \phi_5 &= (\mu_{30} - 3\mu_{12})(\mu_{30} + \mu_{12}) \\ &\quad \times [(\mu_{30} + \mu_{12})^2 - 3(\mu_{30} + \mu_{12})^2] \\ &\quad + (3\mu_{21} - \mu_{03})(\mu_{21} + \mu_{03}) \\ &\quad \times [3(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2], \\ \phi_6 &= (\mu_{20} - \mu_{02}) [(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2] \\ &\quad + 4\mu_{11}(\mu_{30} + \mu_{12})(\mu_{21} + \mu_{03}), \\ \phi_7 &= (3\mu_{21} - \mu_{03})(\mu_{03} + \mu_{21}) [3(\mu_{30} + \mu_{12})^2 - 3(\mu_{21} + \mu_{03})^2] \\ &\quad - (\mu_{30} - 3\mu_{12})(\mu_{21} + \mu_{03}) \\ &\quad \times [3(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2]. \end{aligned} \quad (3)$$

The aforementioned seven parameters  $\phi_1 \sim \phi_7$  are used as identification feature.1 to feature.7 in this paper.

**3.1.2. Texture Features.** Texture features can be used to measure the characteristics of smoothness, roughness, and regularity, and so forth. There are primarily three description methods for texture feature: statistical method, structured method, and spectrum method [13]. GLCM is an excellent statistical method in texture analysis, which can commendably convert gray value into texture information. GLCM is the statistics of probability  $p(x, y)$  of a pair of image elements gray value with certain position relationship. The 14 feature parameters inferred from GLCM have all contained texture information. Nevertheless, they could bring about information redundancy when describing texture of surface defect

Feature sequence	1	2	3	4	5	6	·	·	·	l
Chromosome	1	0	0	1	0	0	·	·	·	1

FIGURE 2: Chromosome coding of feature sequence.

image. This paper chooses the following four parameters with preferable descriptiveness and independence.

(1) Angular second-order moment (energy)

$$A_1 = \sum_x \sum_y p(x, y)^2. \quad (4)$$

(2) Entropy

$$A_2 = -\sum_x \sum_y p(x, y) \log p(x, y). \quad (5)$$

(3) Contrast (inertia moment)

$$A_3 = \sum_x \sum_y m^2 [p(x, y)], \quad m = |x - y|. \quad (6)$$

(4) Relevance

$$A_4 = \frac{\sum_x \sum_y xy p(x, y) - \mu_1 \mu_2}{\sigma_1^2 \sigma_2^2}, \quad (7)$$

where,  $\mu_1$ ,  $\mu_2$ ,  $\sigma_1^2$ , and  $\sigma_2^2$  are, respectively, defined as

$$\begin{aligned} \mu_1 &= \sum_x x \sum_y p(x, y), \\ \mu_2 &= \sum_x y \sum_y p(x, y), \\ \sigma_1^2 &= \sum_x (x - \mu_1)^2 \sum_y p(x, y), \\ \sigma_2^2 &= \sum_y (y - \mu_2)^2 \sum_x p(x, y). \end{aligned} \quad (8)$$

This paper uses the mean value and the standard deviation of the aforementioned four parameters (i.e., energy, entropy, contrast, and relevance), total 8 subfeatures as identification feature.8 to feature.15 in this paper.

**3.2. Adaptive Genetic Algorithm.** Genetic algorithm is an adaptive optimization algorithm simulating biological evolution mechanism [7], which has strong searching capabilities in parallel pattern space and can quickly approach to the global optimal solutions.

*(1) Chromosome Coding and Initial Population Setting.* As shown in Figure 2, if the primary feature sequence extracted from training set contains  $l$  features, define a 0-1 binary code with chromosome length of  $l$ , where  $l = 15$  corresponding to the preceding extracted features. If the  $i$ th chromosome is 1,

the corresponding feature is selected, otherwise not selected. Each chromosome corresponds to a feature subsequence.

Setting  $N$  randomly generates chromosomes as initial population. This paper take the  $N = 1000$  in order to ensure the individual diversity of the population.

(2) *Fitness Function Designing*. On one hand, the larger the feature numbers in the feature subsequence, the more complicated the identification model, which could decrease the last identification performance with greater computational cost and declining antinoise capability.

On the other hand, the identification accuracy can be determined by the total error number  $To_e$  and the false alarm number  $Mi_e$  when the true/false target numbers are already known.

Therefore, the abovementioned three factors must be considered simultaneously when evaluating the feature subsequence from the practical application. The fitness function is defined as

$$F(f_i) = -(n \lg(L) + To_e \lg(To) + Mi_e \lg(Mi)), \quad (9)$$

where  $f_i$  is feature vector corresponding to feature subsequence,  $L$  is the total features number (i.e., the chromosomes length),  $To$  is the total training images number, and  $Mi$  is the true defects number.

From (9), we can see that the fewer the selected features, the total errors, and the false alarm numbers, the larger the received fitness value.

(3) *Genetic Operation*. The individual evolution is finished by genetic operators in GA. In this paper, the initial crossover operator  $op_c = 0.8$ , and the initial mutation operator  $op_v = 0.01$ . This paper adopts two-point crossover method to adaptively adjust the probability of crossover and mutation in the searching process.

The adaptive crossover operator is defined as

$$op_c = \begin{cases} op_{c1} - \frac{(op_{c1} - op_{c0})(F' - F_{av})}{F_{mx} - F_{av}}, & F' \geq F_{av}, \\ op_{c0} - (op_{c1} - op_{c0}) \frac{F'}{F_{av}}, & F' < F_{av}. \end{cases} \quad (10)$$

The adaptive mutation operator is defined as

$$op_v = \begin{cases} op_{v1} - \frac{(op_{v1} - op_{v0})(F_{mx} - F)}{F_{mx} - F_{av}}, & F \geq F_{av}, \\ op_{v0} - (op_{v1} - op_{v0}) \frac{F}{F_{av}}, & F < F_{av}, \end{cases} \quad (11)$$

where  $F'$  is the larger fitness value of the two crossover individuals and  $F_{mx}$  and  $F_{av}$  are the largest fitness value and the average fitness value separately.

(4) *Termination Conditions of AGA*. The AGA is terminated when the iteration times of AGA are equal to the maximum iteration times  $MxGe$ . In this paper, we set  $MxGe$  to 400.

The algorithm flow chart is shown in Figure 3.

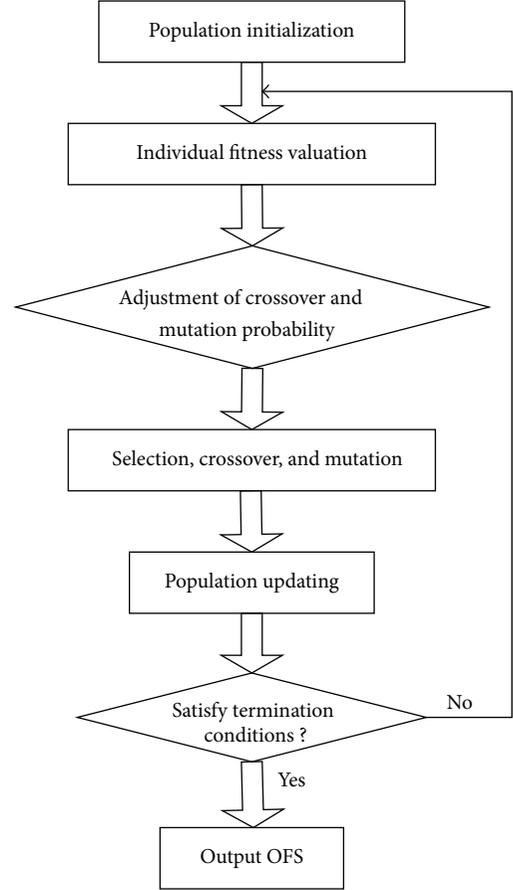


FIGURE 3: The feature selection algorithm flow chart based on GA.

The specific algorithm steps are as follows (where  $Ge$  is the iterations):

Population initialization:  $Co \leftarrow$  randomly generating  $N$  chromosomes with length of  $k$ ;  
while ( $Ge \leq MxGe$  or  $op_v \leq 0.09$ ), do

- (1) fitness valuation: figure out  $F(x)$  for each individual  $x$  in  $Co$ ;
- (2) adaptive adjustment of  $op_c$  and  $op_v$ : using two-point crossover method to any two individuals;
- (3) selection, crossover, mutation;
- (4) population updating:  $Co \leftarrow Co_n$ ;

end  
output the OFS from  $Co$ .

3.3. *Feature Saliency Measuring*. This paper optimizes the selected features by measuring feature saliency in accordance with probabilities of each feature in features space. Suppose that the OFS is  $f_o = [f_{o1}, f_{o2}, \dots, f_{om}]$  after using AGA; then make a certain feature  $f_{oi} \in f_o$  input into SVM for classification. The total accuracy rate is

$$P_{ri} = \frac{To - To_{ei}}{To}. \quad (12)$$

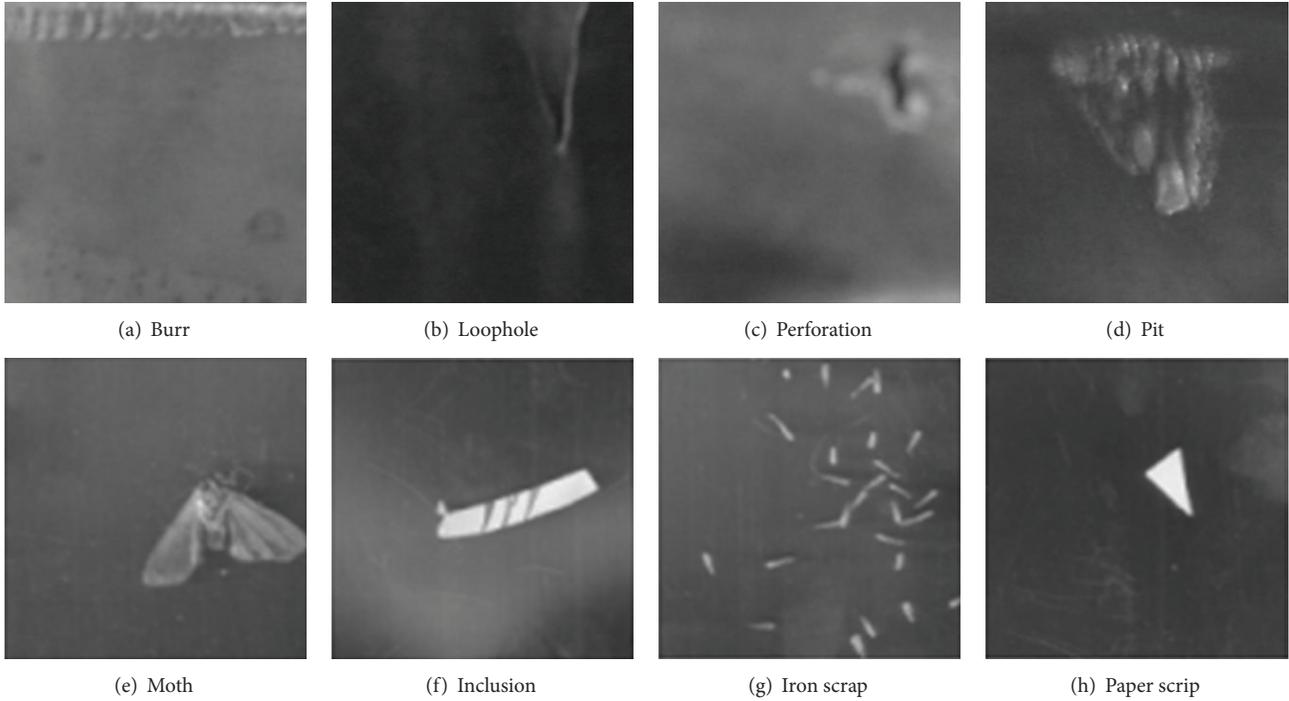


FIGURE 4: Defect samples.

The target accuracy rate is

$$P_{tri} = \frac{Mi - Mi_{ei}}{Mi}. \quad (13)$$

The contribution degree  $D_i$  of feature  $f_{oi}$  is defined as

$$D_i = \frac{1}{2} (P_{ri} + P_{tri}). \quad (14)$$

Then, the normalization is applied to the contribution degree  $D_i$ , and the weight of feature  $f_{oi}$  is defined as

$$W_i = \frac{D_i}{\sum_{i=1}^m D_i}. \quad (15)$$

**3.4. Basic Theory of SVM.** Support vector machine (SVM) [10] is a learning method, which is based on structural risk minimization (SRM) rules. It means that the indicator function set  $S$  is decomposed into several subsets function sequence  $S_1 \subset S_2 \subset \dots \subset S_n \subset S$ , arranging the subsets according to the VC dimension size. After weighing the subset of the empirical risk and confidence range, we will obtain the smallest actual risk. It combines the theory of maximum interface classifier with the method based on kernel, and it can achieve the globally unique optimum solution. Besides, it has unique advantages in small samples, nonlinear problem, and high-dimensional pattern recognition problem. The basic idea of support vector machine is transforming the input space to a high-dimensional space by nonlinear conversion defined by inner product function firstly. In SVM, the choice of the appropriate kernel functions is an important factor, and different kernel function can form different algorithms, which will directly influence the generalization ability and error

control of SVM. Gaussian radial basis kernel, polynomial kernel, B-spline kernel, Fourier kernel, and sigmoid kernel are frequently used kernel functions.

In this paper, the one-versus-one method is chosen to build the SVM classifier, and the polynomial kernel is used as the kernel function.

## 4. Experimental Results and Discussion

In this section, the authors present the experiment results followed by previous approaches on copper strips that are provided by XINGRONG manufacture corporation in Changzhou, Jiangsu province, China to evaluate the performance of the proposed identification method. As shown in Figure 4, we choose eight typical defect target images that include true defect samples (burr, loophole, perforation, and pit) and false defect samples (moth, inclusion, iron scrap, and paper scrip). To extend the image database, we rotate the image sample and transform it in different resolutions. After that, we randomly choose 214 true defect images and 362 false defect images as the training set; the remaining 153 true defect images and 423 false defect images are used as testing set.

**4.1. Feature Extraction.** Figure 5 shows the spatial distribution map of each extracted feature of the defect images. From the maps, we can see that each feature can reflect the differences between true defects and false defects in a certain degree.

### 4.2. Feature Selection

**(1) Single Feature.** Table 1 shows the identification results using single feature. Figure 6 corresponds to Table 1. Table 1

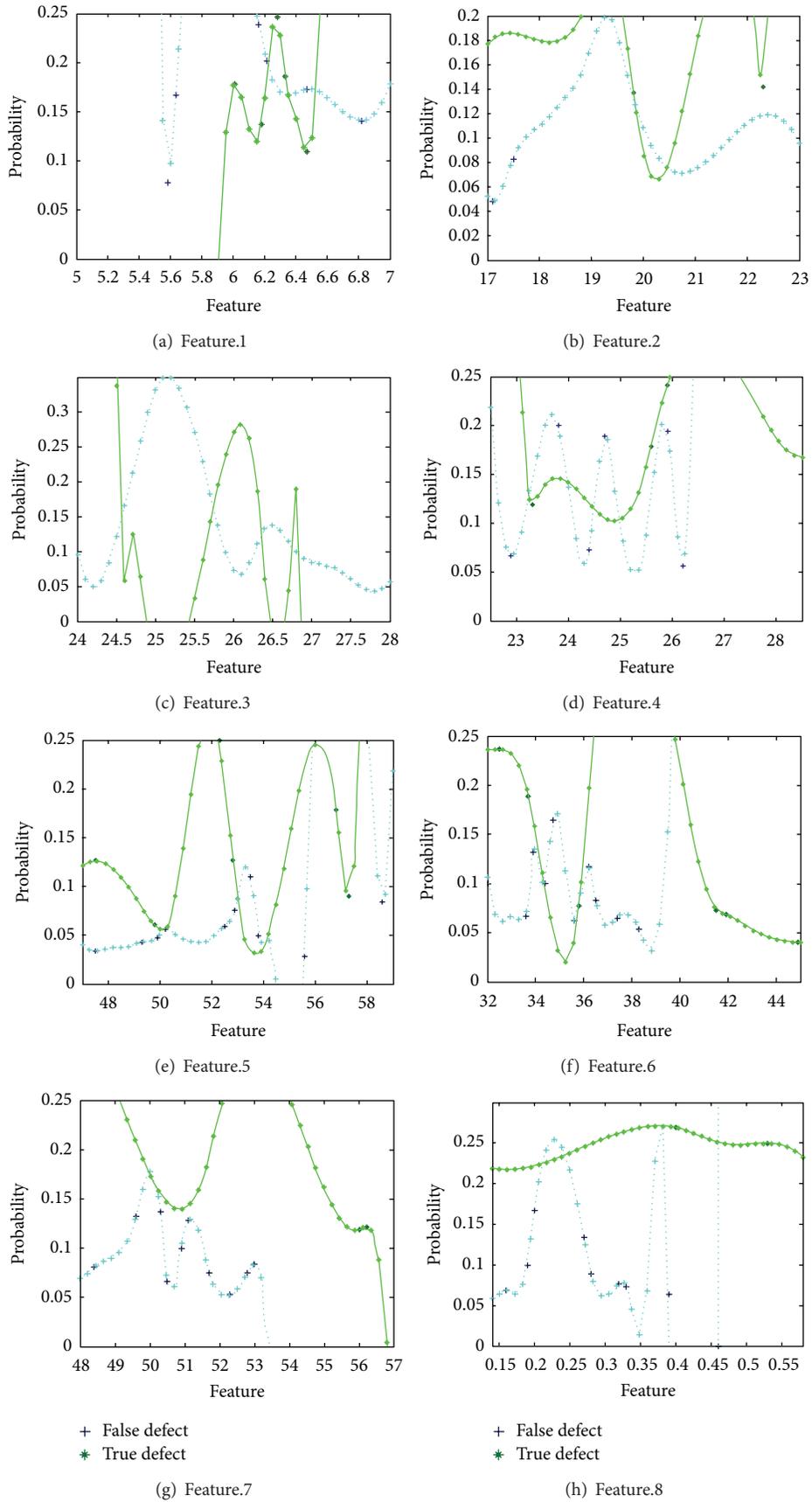


FIGURE 5: Continued.

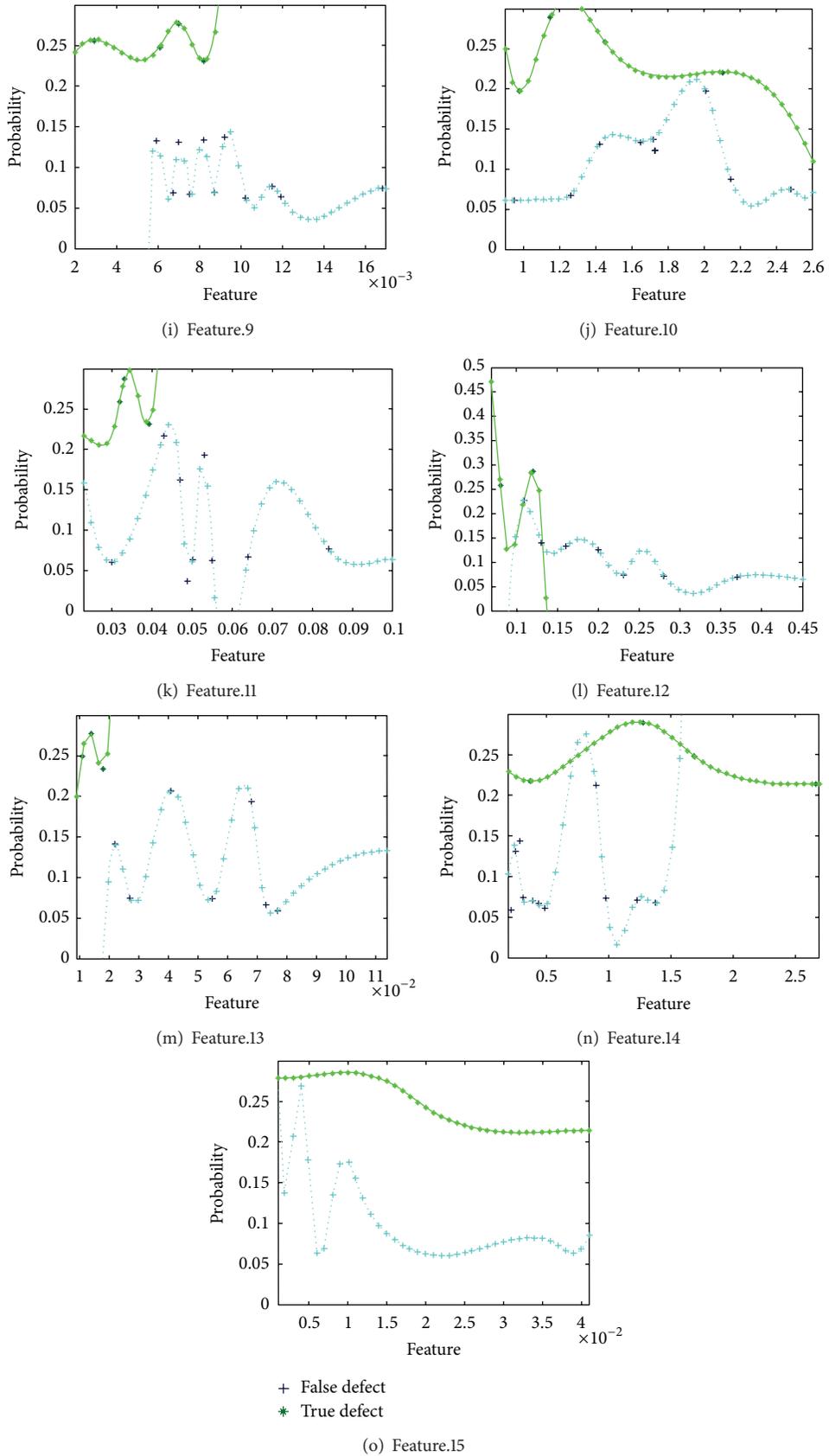


FIGURE 5: Spatial distribution maps of features.

TABLE 1: Identification accuracy rates using single feature.

Feature	Training set		Testing set	
	Total correct rate	Target correct rate	Total correct rate	Target correct rate
f1	0.8056	0.7533	0.8554	0.7727
f2	0.5637	0.4801	0.5373	0.3359
f3	0.6119	0.5003	0.6883	0.4988
f4	0.6715	0.7092	0.7004	0.5013
f5	0.5508	0.3342	0.6017	0.3081
f6	0.7702	0.6547	0.8009	0.6833
f7	0.8113	0.7087	0.8351	0.7104
f8	0.7518	0.6476	0.7991	0.6506
f9	0.8683	0.7812	0.8790	0.7941
f10	0.6013	0.4747	0.5706	0.3987
f11	0.8896	0.8900	0.8711	0.9014
f12	0.9004	0.9151	0.9107	0.8992
f13	0.9417	0.9450	0.9625	0.9633
f14	0.8006	0.7229	0.8513	0.7094
f15	0.5725	0.3918	0.5631	0.3614

TABLE 2: Results for AGA in 10 times.

Times	OFS coding	Feature numbers	Fitness value	Consuming time (s)
1	[100001111011110]	9	-9.0103	64.03
2	[100001111011110]	9	-9.0103	74.67
3	[100001111011110]	9	-9.0103	70.79
4	[100001111011110]	9	-9.0103	72.88
5	[100001111011110]	9	-9.0103	69.53
6	[100001111011110]	9	-9.0103	68.96
7	[100001111011110]	9	-9.0103	79.25
8	[100001111011110]	9	-9.0103	61.94
9	[100001111011110]	9	-9.0103	56.78
10	[100001111011110]	9	-9.0103	67.41
Average	[100001111011110]	9	-9.0103	68.62

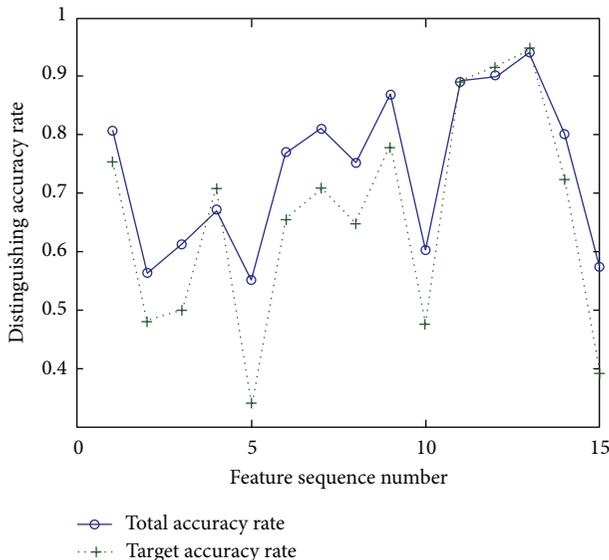


FIGURE 6: Identification accuracy rates using single feature.

and Figure 6 further illustrate that each of the 15 features extracted in this paper has a certain identification ability for the true and false defects.

(2) *Multifeature.* As shown in Table 2, the selected OFS using AGA in 10 times is the same, that is  $[f_1, f_6, f_7, f_8, f_9, f_{11}, f_{12}, f_{13}, f_{14}]$ . The corresponding fitness value is  $-9.0103$ , and the average consuming time is 68.62 s, which interprets that the processing speed of the proposed method is very fast.

Table 3 shows the experimental results using the selected OFS inputted into SVM. The total error using the OFS for training set is 9, and the false alarm number is 3; thus, the total is 0.9817, and the target correct rate is 0.9939. The total error using the OFS for testing set is merely 5, and the false alarm number is just 1; thus, the total correct rate is 0.9886, and the target correct rate is 0.9903. Compared with the results in Table 1, we can see that the identification accuracy using OFS is higher than using single feature independently.

Combined with Table 1, the weight of each feature in OFS calculated by (12) to (15) is shown in Table 4.

TABLE 3: Identification accuracy of optimal feature subsequence.

		Training set		Testing set			
Total errors	False alarm	Total correct rate	Target correct rate	Total errors	False alarm	Total correct rate	Target correct rate
9	3	0.9817	0.9939	5	1	0.9886	0.9903

TABLE 4: Weight of each feature in optimal feature subsequence.

	f1	f6	f7	f8	f9	f11	f12	f13	f14
$D_i$	0.7795	0.7125	0.7600	0.6997	0.8248	0.8898	0.9078	0.9434	0.7618
Weight	0.1071	0.0979	0.1044	0.0961	0.1133	0.1222	0.1247	0.1296	0.1047

TABLE 5: Identification accuracy of total feature sequence.

Feature coding: [11111111111111], Fitness value: -19.1774							
		Training set		Testing set			
Total errors	False alarm	Total correct rate	Target correct rate	Total errors	False alarm	Total correct rate	Target correct rate
10	7	0.9796	0.9624	14	13	0.9682	0.8738

In order to compare the performance of the proposed method, we use TFS inputted into SVM classifier under the same condition. And the identification results are shown in Table 5. From Table 5 we can see that the total errors for training set is 10, and the false alarm numbers is 7, thus the total is 0.9796, and the target correct rate is 0.9624. The total error for the testing set is 14, and the false alarm number is 13; thus, the total correct rate is 0.9682, and the target correct rate is 0.8738. By comparing Table 5 with Tables 1 and 3 it is observed that the identification accuracy using TFS for training set is higher than either single feature, but lower than that using OFS. Furthermore, the identification accuracy using TFS for the testing set is far below than using OFS, which further interpreted that the proposed feature selection method increases the identification accuracy rate greatly.

The neural network has been used in pattern recognition widely. Hundreds of defect images are randomly chosen from the database as the training sample; then we choose the other hundreds of images as the test sample. Table 6 shows the identification results using RBF neural network.

From Tables 3, 5, and 6, we can see that the identification accuracy of SVM can be as high as 99.03%, and in most instances this number is more than 95%. However, the identification accuracy of RBF neural network is no more than 95%. It is obvious that the performance of the proposed method is better than the RBF neural network.

### 5. Conclusions

In this study, a new surface defect target identification system for copper strip based on adaptive genetic algorithm and feature saliency was developed. Genetic algorithm has the advantage of fast convergence and can avoid involving into local optimal solution. In the proposed method, the probability of crossover and mutation was adjusted dynamically according to the fitness value, which had been calculated by integrating total error rate and false alarm rate. Furthermore,

TABLE 6: Identification accuracy of RBF neural network.

Defect type	Total	Correct	Accuracy
Pits	126	112	88.9%
Loophole	342	323	94.4%
Burr	215	203	94.4%
Perforation	106	96	90.6%

to evaluate the performance of the proposed method, comparison between the selected feature subsequence and total feature sequence has been implemented. The experimental results demonstrate that the proposed approach increases the correct rate and lowers the false alarm rate. In the proposed method, feature extraction can decrease the dimension of features and can increase the speed of processing. However, crossover and mutation operators in genetic algorithms always work under the condition of a certain probability; it will lead to “degradation” phenomenon inevitably, such as prematurity and species diversity decreasing. Some potential priori knowledge of the actual problem itself cannot be applied in the genetic algorithms. The future work should attempt to verify the robustness and effectiveness of our method in practical application.

### Acknowledgment

This paper is supported by the National Natural Sciences Foundation of China (Grant no. 61273170).

### References

- [1] J.-X. Huang, D. Li, F. Ye, and W. Zhang, “Detection of surface defect of solder on flexible printed circuit,” *Optics and Precision Engineering*, vol. 18, no. 11, pp. 2443–2453, 2010.
- [2] X.-W. Zhang, Y.-Q. Ding, Y. Lv, A. Shi, and R. Liang, “A vision inspection system for the surface defects of strongly

- reflected metal based on multi-class SVM,” *Expert Systems with Applications*, vol. 38, no. 5, pp. 5930–5939, 2011.
- [3] W. Ping, Z. Xuewu, M. Yan, and W. Zhihui, “The copper surface defects inspection system based on computer vision,” in *Proceedings of the 4th International Conference on Natural Computation (ICNC '08)*, pp. 535–539, October 2008.
- [4] A. S. Tolba, H. A. Khan, A. M. Mutawa, and S. M. Alsaleem, “Decision fusion for visual inspection of textiles,” *Textile Research Journal*, vol. 80, no. 19, pp. 2094–2106, 2010.
- [5] W. Wang and Q.-H. Zheng, “Feature selection for text categorization using filtering and wrapping,” *Journal of Computational Information Systems*, vol. 2, no. 4, pp. 1333–1342, 2006.
- [6] I. Rodriguez-Lujan, R. Huerta, C. Elkan, and C. S. Cruz, “Quadratic programming feature selection,” *Journal of Machine Learning Research*, vol. 11, pp. 1491–1516, 2010.
- [7] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, Boston, Mass, USA, 1989.
- [8] W.-Z. Yang, D.-I. Li, and L. Zhu, “An improved genetic algorithm for optimal feature subset selection from multi-character feature set,” *Expert Systems with Applications*, vol. 38, no. 3, pp. 2733–2740, 2011.
- [9] H. Zhang, R. Tao, Z. Li, and H. Du, “A feature selection method based on adaptive simulated annealing genetic algorithm,” *Binggong Xuebao/Acta Armamentarii*, vol. 30, no. 1, pp. 81–85, 2009.
- [10] C. Cortes and V. Vapnik, “Support-vector networks,” *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.
- [11] A. Papoulis, *Probability, Random Variables and Stochastic Processes*, Mc Graw-Hill Kogakusha, Akita, Japan, 1965.
- [12] M. Hu, “Visual pattern recognition by moment invariants,” *IRE Transactions on Information Theory*, vol. 8, no. 2, pp. 179–187, 2002.
- [13] Y. Chiou, C. Lin, and B. Chiou, “The feature extraction and analysis of flaw detection and classification in BGA gold-plating areas,” *Expert Systems with Applications*, vol. 35, no. 4, pp. 1771–1779, 2008.

## Research Article

# Research on the Production Scheduling Optimization for Virtual Enterprises

Min Huang,<sup>1</sup> Ruixian Huang,<sup>1</sup> Bo Sun,<sup>2</sup> and Linrong Li<sup>1</sup>

<sup>1</sup> School of Software Engineering, South China University of Technology, Guangzhou 510006, China

<sup>2</sup> Department of Research, Guangdong University of Foreign Studies, Guangzhou 510420, China

Correspondence should be addressed to Min Huang; [minh@scut.edu.cn](mailto:minh@scut.edu.cn)

Received 12 March 2013; Accepted 2 July 2013

Academic Editor: Orwa Jaber Housheya

Copyright © 2013 Min Huang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Production scheduling is a rather difficult problem in virtual enterprises (VE) for the tasks of production which would be executed by some distributed and independent members. Factors such as the timing constraints of task and ability restrictions of the members are considered comprehensively to solve the global scheduling optimization problem. This paper establishes a partner selection model based on an improved ant colony algorithm at first, then presents a production scheduling framework with two layers as global scheduling and local scheduling for virtual enterprise, and gives a global scheduling mathematical model with the smallest total production time based on it. An improved genetic algorithm is proposed in the model to solve the time complexity of virtual enterprise production scheduling. The presented experimental results validate the optimization of the model and the efficiency of the algorithm.

## 1. Introduction

Virtual enterprise (VE) is a dynamic alliance consisting of independent enterprises. Production scheduling is one of the most important aspects in VE to seize the market opportunities with the shortest production cycle and maximum profits. Hence, production planning has become an important area in VE.

As Helaakoski et al. defined, VE is defined as a temporary collaborative network consisting of independent enterprises, formed to exploit a particular business opportunity [1]. Goel et al. look into both the paradigms and Enterprise Architecture viewpoint of virtual enterprise. Enterprise Architecture deals with the structure of an enterprise, relationships, and interactions of its units [2]. Capuano et al. propose an overview of the Knowledge Virtual Enterprise model, where the virtual enterprise vision is extended with Knowledge-based assets in order to provide an agreement model to support the interoperability among organizations [3]. Danesh et al. propose a 6-layer framework with multiple components within each layer and present a distributed SOA infrastructure that facilitates peer-to-peer collaboration between

enterprises in a virtual enterprise [4]. Esparcia et al. present an extension of the Environment Dimension of the VE model, which is an Organization Modeling Language to define Organization-Centered Multiagent Systems [5].

Partner selection of virtual enterprise is important to VE and production scheduling of VE. Crispim and Sousa mainly talk about the procedure of partner selection in VE and give a model but cannot solve it absolutely [6]. Jarimo and Salo create a mixed-integer linear programming (MILP) model to solve partner selection in virtual organization [7]. In additionally to fixed and variable costs, we present extensions that accommodate transportation costs, capacity risk measures, and interorganizational dependencies such as the success of past collaboration. Nayak et al. propose a variant of swarm optimization to handle combinatorial problems efficiently compared to its continuous counterpart [8]. Simona and Raluca propose partner selection as a difficult task and involve important decision-making because it includes many factors: quality, cost, trust, delivery time, limitations of geography, and communicate abilities [9]. At last, they solve the problem by genetic algorithm. Mohamed focuses on the solution procedure of the multiobjective

partner selection problem in virtual enterprise where the cost coefficients are expressed as interval by the decision maker and uses a multiobjective algorithm (PSA) to solve [10].

Yalaoui et al. solve a hybrid flow shop scheduling problem and create a new method to solve the problem based on the nature which is the particle swarm optimization method under fuzzy logic controller (FLCPSO) [11]. Dugardin et al. create a model to solve multiobjective scheduling of a reentrant hybrid flow shop problem and can be used for partner selection in VE if improved [12]. Gao and Jiang and Ding et al. establish a mathematical model solved using a hybrid genetic algorithm (GA) to acquire the shortest operating cycle based on the characteristics of the production scheduling of VE [13, 14]. However, this model caused premature convergence because that it only used a single population. Song combines the widely applied “cloud computing” theory to present the cloud of VE production planning and control model [15]. However, the model has not yet been tested in practical applications. Gao and Ding, and Li and Liu discuss an order rarely multiple orders, established a multiorder production scheduling model, and introduce various population genetic-simulated annealing algorithms to solve this model [16, 17]. Although the problem of population diversity is solved, the execution time is very slow. Some researchers such as Camarinha-Matos and Afsarmanesh and Zhao and Zhou apply multiagent technology to solve VE production planning and control [18, 19]. They mainly attempt to establish a framework of the production scheduling system. The VE production planning model and the corresponding algorithm need to be further studied. According to the two-tier scheduling model of VE, Tao and Xie establish a mathematical model with the smallest total operating time and used the ant colony algorithm to solve the model [20]. However, premature convergence and slow execution time were the disadvantages of this model.

The structure of the paper is as follows. Section 2 states the problems of partner selection and production planning in VE and creates two models. Section 3 introduces improved ant colony algorithm (IACO) to solve partner selection model and improved multipopulation genetic algorithm (IMGA) to solve the production scheduling model. Section 4 presents a numerical simulation to show the feasibility of the algorithm. Finally, our solution to the problem and the performance of IMGA are discussed. Section 5 concludes this paper and ongoing works.

## 2. Description of the Problem and Model

### 2.1. Partner Selection Model

**2.1.1. Three-Stage Model of Partner Selection.** According to the actual characteristics of the virtual enterprise partner selection and the cycle of the selection, the virtual enterprise partner selection can be roughly divided into three stages as the primary selection, fine selection, and optimized combination, as shown in Figure 1.

Primary and fine selection stage is relatively simple; generally the dominated enterprise establishes a specific

assessment team and filters based on certain indicators, excluding the enterprises which do not have the required core resources. Combinatorial optimization phase occupies an important position in the entire virtual enterprise partner selection, and it is related to the success or failure of the partner selection. Therefore, this paper focuses on the establishment of the stages of the model and use of improved ant colony algorithm to solve it.

### 2.1.2. Partner Selection Combinatorial Optimization Model

**Description of the Problem.** Assume that the dominant enterprise has decomposed the task into  $J$  different subtasks based on the decomposition of business processes. After the primaries of the first phase and the fine selection of the second phase, we can get the combination of the candidate partner enterprises set  $E^{I \times J}$  that need to optimize, where  $J$  represents the task number and  $I$  is the number of candidate enterprises for each task.  $u_{ij}$  is task  $j$  which selects its candidate partners in the corporate collection of the  $i$ th enterprise as the final implementation of the enterprise,  $i = 1, 2, \dots, I, j = 1, 2, \dots, J$ .

Due to the difference of the factor of the characteristics of the market opportunities and core enterprise defects, consideration of core enterprise during the partner selection is also different. Overall, the time, costs and risk are factors to be considered basically in every virtual enterprise [21]. Therefore, this paper considers the impact of the virtual enterprise partner selection factors from three aspects of time, cost, and risk.

**Determination of the Objective Function.** According to the characteristics of the virtual enterprise partner selection, set the time  $T$ , the cost  $C$ , and the risk  $R$  the three objective functions as follows:

$$\min T = \min \left[ \sum_{j=1}^J \sum_{i=1}^I (T_{ij}H_{ij}) \right], \quad (1)$$

$$\min C = \min \left[ \sum_{j=1}^J \sum_{i=1}^I (C_{ij}H_{ij}) \right], \quad (2)$$

$$\min R = \min \left[ \sum_{j=1}^J \sum_{i=1}^I (R_{ij}H_{ij}) \right], \quad (3)$$

$$H_{ij} = \begin{cases} 1, & \text{the candidate partner } u_{ij} \text{ is selected,} \\ 0, & \text{the candidate partner } u_{ij} \text{ is not selected.} \end{cases} \quad (4)$$

Wherein  $T_{ij}$  indicates the required time when the task  $j$  is the  $i$ th candidate corporate executive,  $C_{ij}$  represents required costs when task  $j$  is the  $i$ th candidate corporate executive,  $R_{ij}$  is the risk that chooses the  $i$ th candidate enterprises to execute task  $j$ .  $J$  represents the total number of tasks;  $I$  is the number of candidate enterprises for the  $j$ th task. Ultimate goal is a minimum time, cost, and risk.

For multiobjective decision-making problems, it is difficult for the decision makers to prepare, give the specific means, and consider the polarity inconsistency of goal.

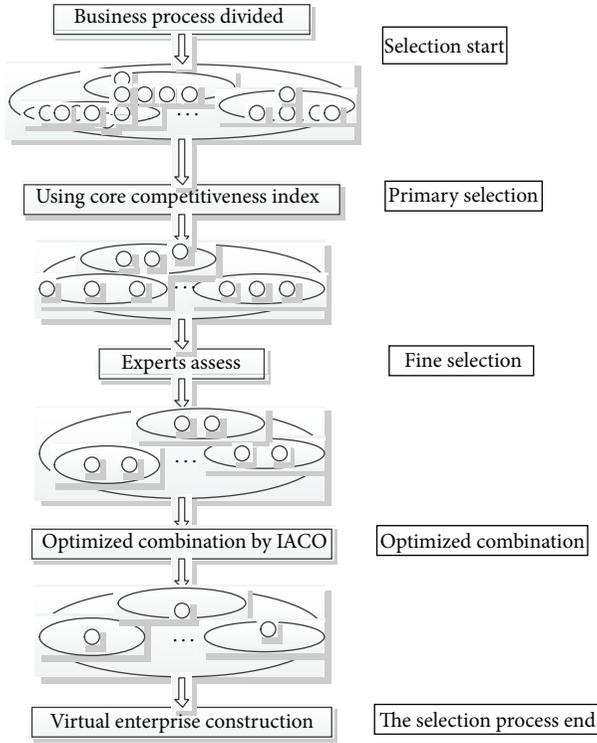


FIGURE 1: Three-stage model of partner selection.

Ideal point method (TOPSIS) is a good method for solving multiobjective decision problems. Guo and Jin applied TOPSIS to solve the problem of multiple index decision grey relation [22]. Ideal point method was first proposed by C. L. Hwang and K. Yoon in 1981 [23], sort ideal point method sort the closeness  $e$  based a finite number of evaluation objects with idealized goal, and the relative merits of existing object evaluation ideal point method is an effective method multiobjective decision analysis, also known as pros and cons of solutions of distance method. Define a measure in space to measure the degree of a program close to the ideal solution and away from negative ideal solution, pros and cons of sort of candidate programs based on the value of this measure to provide a basis for decision-making. This method is based on obtaining a good application.

Use ideal point method, and construct the fitness function of the multiobjective decision problems; decision makers need to give positive ideal value (positive ideal point) of each goal and negative ideal value (negative ideal point). The positive ideal point of the time, cost, and risk in this paper is  $(T^+, C^+, R^+)$ , negative ideal point is  $(T^-, C^-, R^-)$ . This can be constructed out the objective function  $f(t)$ .

$$f(t) = \frac{d^-}{d^- + d^+}, \quad (5)$$

$$d^-(t) = \left\| \left( \frac{T(t) - T^-}{T^+}, \frac{C(t) - C^-}{C^+}, \frac{R(t) - R^-}{R^+} \right) \right\|, \quad (6)$$

$$d^+(t) = \left\| \left( \frac{T(t) - T^+}{T^-}, \frac{C(t) - C^+}{C^-}, \frac{R(t) - R^+}{R^-} \right) \right\|, \quad (7)$$

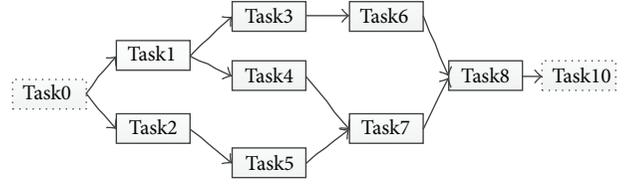


FIGURE 2: The figure of task decomposition.

where  $\| \cdot \|$  means to take norm,  $t$  is the number of ants,  $d^+$  is the positive ideal distance measure, and  $d^-$  point is the negative ideal distance measure. The Euclidean norm can be used to solve it, and the values  $d^+$  and  $d^-$  can be calculated using

$$d^+(t) = \sqrt{(T(t) - T^+)^2 + (C(t) - C^+)^2 + (R(t) - R^+)^2},$$

$$d^-(t) = \sqrt{(T(t) - T^-)^2 + (C(t) - C^-)^2 + (R(t) - R^-)^2}. \quad (8)$$

**2.1.3. Numerical Example.** As shown in Figure 2, it is assumed that the dominated enterprise decomposition market opportunities objectives into Task1 ... Task8. Task0 and Task9 are the starting point and end point.

Through the first phase and the second phase of the selecting, the number of candidate enterprises which can enter into the optimum combination of stage has been narrowed into an appropriate range. The information set of Task1 ... Task8 in optimized combination stage corresponding to candidate companies is shown in Table 1.

According to the information provided in Table 1, calculate each candidate set using the ideal point method. For example, positive ideal points of the candidate set  $\{P_{11}, P_{12}, P_{13}, P_{14}\}$  is  $(90, 5.5, 0.35)$ , and the negative ideal point is  $(110, 7.5, 0.45)$ . The calculated ant path hunt to the diagram shown in Figure 3 is carried out for all candidate sets. Calculating the collection of all candidates can get the digraph of ants' path finding.

## 2.2. Production Scheduling Model

**2.2.1. Description of the Problem.** After the dominant enterprise establishes VE, the  $N$  subtasks are assigned to the  $M$  member enterprises. The production plan is developed based on the timing relationships between the tasks and the constraints of the overall scheduling. This model aims at obtaining the maximum profits and deliver customer-specified products with the shortest production cycle [24]. In this paper, we consider that the VE production planning has the following characteristics [25].

- The dominant enterprise specifies the operating time of the decomposition of the task.
- Subtasks in the implementation process without interruption, for author/s of only one affiliation.
- Subtasks with timing constraints.
- When member enterprises are assigned to multiple tasks, considering the member enterprises of their

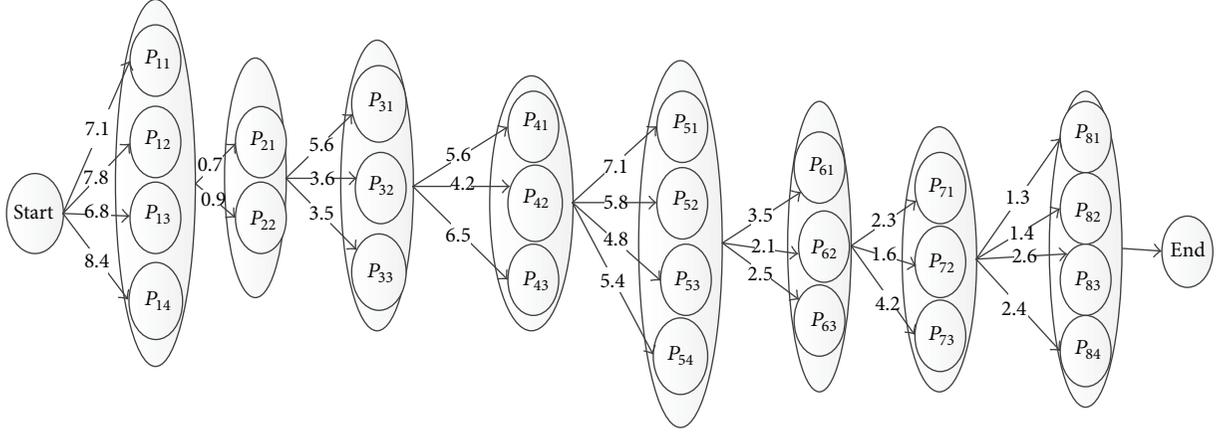


FIGURE 3: Digraph with weight.

TABLE 1: Time, cost, and risk of candidate enterprises.

Task	Candidate enterprise	Cost (10000\$)	Time (month)	Risk
Task1	$P_{11}$	100	6.5	0.41
	$P_{12}$	99	7.1	0.40
	$P_{13}$	110	5.5	0.35
	$P_{14}$	90	7.5	0.45
Task2	$P_{21}$	56	2.7	0.22
	$P_{22}$	77	3.3	0.33
Task3	$P_{31}$	59	5.3	0.50
	$P_{32}$	67	5.5	0.45
	$P_{33}$	50	4.5	0.35
Task4	$P_{41}$	80	5	0.33
	$P_{42}$	88	3.2	0.51
	$P_{43}$	79	3.4	0.52
Task5	$P_{51}$	130	5.2	0.22
	$P_{52}$	139	4	0.33
	$P_{53}$	122	6.2	0.25
	$P_{54}$	140	4	0.20
Task6	$P_{61}$	80	5	0.33
	$P_{62}$	79	3.4	0.52
	$P_{63}$	88	3.2	0.51
Task7	$P_{71}$	56	2.7	0.22
	$P_{72}$	77	3.3	0.22
	$P_{73}$	103	4.1	0.25
Task8	$P_{81}$	59	5.3	0.50
	$P_{82}$	67	5.5	0.45
	$P_{83}$	50	4.5	0.35
	$P_{84}$	66	4	0.25

own resource constraints, a moment can only perform one task.

- (e) Initial time of the production scheduling is 0, and the goal is the shortest operating cycle.

2.2.2. *Symbol Description.* This paper uses the following symbols to describe the mathematical model of VE:

$M$ : the total number of tasks;

$N$ : the total number of enterprises;

$P_i$ : task number,  $i \in [1, N]$ ;

$E_i$ : the number of enterprises,  $i \in [1, M]$ ;

$R_i$ : enterprise  $E_i$  assuming the task set;

$K_i$ : the number of tasks to be undertaken by the enterprise  $E_i$ ;

$st_{ie}$ : the initial production time of  $P_i$  in the enterprise  $E_e$ ;

$ct_{ie}$ : production time of  $P_i$  in the enterprise  $E_e$ ;

$et_{ie}$ : final time of  $P_i$  in the enterprise  $E_e$ ;

$tt_{mn}$ : transport time from enterprise  $E_m$  to  $E_n$ ;

$$x_{ie} = \begin{cases} 1, & \text{task } P_i \text{ in the enterprise } E_i \\ 0, & \text{otherwise;} \end{cases} \quad (9)$$

$$f_{ij} = \begin{cases} 1, & \text{task } P_i \text{ is the preamble of the task of } P_j \\ 0, & \text{otherwise;} \end{cases}$$

$L_x$ : extending task, and  $x \in [1, N]$ ;

$S$ : delaying the task  $L_x$  as a preorder or in connection with tasks performed in the same enterprise;

$S'$ : completed tasks including delayed tasks.

2.2.3. *Model.* Under the constraints, with the initial time of the production scheduling set to 0, the entire production cycle is ensured to be the shortest for each production task in the VE production planning problems by determining the shortest completion time [26]. Delayed by the duration of the task prior to the completion of this task, the subsequent

sequence of tasks and the same enterprise to complete the task to a collection of tasks in  $S$  cannot be executed. The tasks in  $S$  can be reallocated by the dominant enterprise or adjusted dynamically in accordance with the original distribution. VE can be dynamically adjusted according to the sign constraints. The production scheduling model is expressed as follow:

$$\min(T) = \min \left\{ \max_{1 \leq i \leq N} \max_{1 \leq e \leq M} et_{ie} \right\}. \quad (10)$$

The constraints of the initial and final times of a task are shown

$$et_{ie} = st_{ie} + ct_{ie}. \quad (11)$$

The task execution time constraints with direct sequence relationship are expressed in

$$(et_{im} - st_{jn} - tt_{mn}) \cdot f_{ij} \cdot x_{im} \cdot x_{jn} \geq 0. \quad (12)$$

As shown in (13), the same task cannot be performed within different enterprises

$$R_m \cap R_n = \emptyset, \quad \forall m, n \in [1, M]. \quad (13)$$

As shown in (14), all subtasks are allocated to the enterprise

$$R_1 \cup R_2 \cup \dots \cup R_M = P_1 \cup P_2 \cup \dots \cup P_N. \quad (14)$$

The time constraints between tasks before and after in the same enterprise are expressed in

$$(st_{je} - st_{ie}) f_{ij} \geq 0. \quad (15)$$

As shown in (16), the initial time of each task cannot be negative

$$st_{ie} \geq 0, \quad \forall i \in [1, M], \forall e \in [1, M]. \quad (16)$$

As shown in (17), each enterprise can only perform one task at a time

$$\sum_{i=1}^{K_n} x_{ie} = 1, \quad \forall e \in [1, M]. \quad (17)$$

As shown in (18), the finished and unfinished tasks are the sum of all the tasks

$$S \cup S' = P_1 \cup P_2 \cup \dots \cup P_N. \quad (18)$$

### 3. Solving the Models by IACO and IMGAs

**3.1. Solving Partner Selection Model by IACO.** Through the above analysis, the use of improved ant colony algorithm (IACO) for virtual enterprise partner selection problem is to use the ants traverse set of candidate companies for each task, each time after only a candidate for a collection of tasks in a candidate enterprises. Improved ant colony algorithms for virtual enterprise partner selection concrete steps are as follows.

**3.1.1. Initialization.** Assume that the core enterprise decomposes the market opportunity goal into  $m$  tasks, and there is a collection of selected partners for each task. The initial number of ants is  $n$ , ants carry the initial pheromone as  $\tau_{ij}(0) = \tau_0$ , ( $\tau_0$  is a constant). The initial size of the pheromone between the enterprise node is obtained by the formula (5). The required value of the reciprocal is called the degree of attraction in this article. For the initial placed node of each ant is set randomly, so the starting point of each ant is not necessarily the same.

**3.1.2. Ants Jump.** Ants exist in discrete states. They use probability transfer rules to move from one node to another node. The path selection probability of ant  $t$  from node  $i$  moving to node  $j$  is

$$P_{ij}^k(t) = \begin{cases} \frac{\tau_{ij}^\alpha(t) \varphi_{ik}^\beta(t)}{\sum_{s \in \text{allowed}_k} [\tau_{is}(t)]^\alpha \cdot [\varphi_{is}(t)]^\beta}, & \text{if in allowed}_k \\ 0, & \text{else,} \end{cases} \quad (19)$$

where  $\tau_{ij}(t)$  represents the concentration of pheromone of the edge  $(i, j)$  at the  $t$ th search cycle,  $k$  is the attraction of  $j$  point (obtained by the inverse of the value of (5)), parameters  $\alpha, \beta$  represent relative weights, for adjusting the relative importance of  $k$  and  $j$ , and the larger  $\alpha$ , the more ants tend to choose the road section that other ants use, reflecting the collaboration among ants. The greater  $\beta$  represents the more influence to the degree of attraction of jump probabilities, and the probabilities are close to the greedy rule. The  $j$  ( $j \in \text{allowed}_k$ ) denotes the set of ant  $k$  that is allowed to choose the next node, that taboo  $[j] = 1$ . If you simply transfer the maximum probability, it will soon fall into local optimum; the optimal solution cannot be found. Therefore, we choose the roulette algorithm which combines the probability to guide the transfer of ants.

**3.1.3. Update Taboo Table.** For each ant can only access the nodes which haven't been visited before, so a taboo table  $[m][n]$  is set to mark it, in which the value of taboo  $[k][t]$  refers the point accessed by ant  $k$  at the time  $t$ . In this algorithm, one task can only be assigned one partner, so when a node is accessed by ants, all of the other nodes in the collection of the corresponding candidate enterprises would be marked as accessed.

**3.1.4. Update Pheromone.** Ants traverse a candidate enterprise collection to construct a feasible solution, that needs to update the global pheromone. The pheromone update includes two aspects: the pheromone that ants leave and volatile pheromone over time. Ants will leave a certain amount of pheromone on the edge of the path, and the size of the pheromone left by ant  $k$  is calculated by the formula (20) as follows:

$$\Delta \tau_{ij}^k(t) = \frac{Q}{L[k]}, \quad (20)$$

where  $Q$  is a positive constant number, according to the specific circumstances.  $L[k]$  is the length of the path that ant  $k$  traveled through; in this paper, the length is calculated by the degrees of the attraction of each other node size plus 1. At the end of the  $t$ th search cycle, the pheromone on the concentration increment remains in the edge  $(i, j)$  as shown

$$\Delta\tau_{ij}(t) = \sum_{k=1}^m \Delta\tau_{ij}^k(t). \quad (21)$$

Pheromone will disappear gradually over time. Assuming that the pheromone retention factor is  $k$ , and the volatiles ratio is  $j$ , which represents the degree of disappearance of the pheromone. So the edge  $(i, j)$  on the global pheromone update is carried out in accordance with

$$\tau_{ij}(t+1) = \rho \cdot \tau_{ij}(t) + \Delta\tau_{ij}(t), \quad \rho \in (0, 1). \quad (22)$$

After (2), (3), and (4), the ants completed a full traversal; after getting an optimal solution, update the global pheromone. Repeated (2), (3), and (4) operations in accordance with the initial number of ants, and finally choose one path that ant go through most as the optimal solution of the algorithm execution.

Using example of part II, the IAOC is run in MATLAB 7.0; the number of ants in initialization is 30, the constants  $Q = 20$ , and  $C = 0.1$ , and the weight coefficients  $\alpha, \beta$  are 0.7 and 0.9. After all ants finished, 27 ants converge to the path (P14, P22, P31, P43, P51, P61, P73, and P83). So we can select the candidate as the ultimate corporate partners on this path. And the candidate corporate on this path can be selected as the ultimate partners.

In the case of large solution space, the algorithm will not converge and thus cannot get the optimal combination of the optimal solution. We solve this problem through adjust the values of  $\alpha, \beta$ , and ants' initial phenomenon.

### 3.2. Solving Scheduling Model by IMGGA

**3.2.1. Encoding and Decoding.** For simple operation, genetic encoding is based on task numbers. For tasks numbered as  $1 - N$ , genetic sequences are the permutations of the  $N$  numbers. Decoding is the sequence of topological sorting. Therefore, genetic sequences are transformed topologically following the constraint graph of the task.

#### 3.2.2. Genetic Operator

**Population Initialization.** A population size of `pop_size` is randomly generated and then randomly allocated to  $n$  sub-populations. The size of sub-population is `pop_size/n`.

**Improved Multipopulation Genetic Algorithm.** The multipopulation GA improves the performance of GA. This paper designs an IMGGA [27] based on the existing multi-population GA. The entire population consists of a more general population and an excellent population, with a good population of the ordinary population evolution of a certain algebraic cross to guide the evolution of the general population and

improve the quality of the general population. The excellent population continues to absorb the best individual from the general population to improve its quality and hasten the convergence rate of GA.

**Fitness Function.** The production model of this paper is based on the shortest production time for the target. Thus, the fitness function is defined as  $f_i = F_{\max j} - F_{ij}$ , where  $F_{\max j}$  is the maximum execution time of all the genes in the sub-population  $j$  and  $F_{ij}$  is the execution time of gene  $i$  in the sub-population  $j$ .

**Select Operator.** In this paper, the general populations are selected using the roulette wheel selection operator and the elitist strategy to ensure that the highest fitness individuals are copied to the next generation.

**Crossover Operator.** The partially mapped crossover is a good method for increasing the diversity of the offspring through the intersection between the parent individuals. Two cross-points are randomly selected, and the fragments between the cross-points of the parent individuals are exchanged. A conflict occurs if the gene outside the cross-point does not crossover the gene fragment. Otherwise, the genes are determined through gene mapping.

**Mutation Operator.** Mutation in combinatorial optimization problems is commonly used in reverse swap and insert operation. The so-called reversion is reversing two different random position genes in the chromosome [28]. In this paper, the reverse operation is used to ensure that the largest possible chromosomal variation is obtained.

**Simulated Annealing Mechanism.** In simulated annealing algorithm [29], an initial solution is provided. Another solution is randomly generated from the field. The acceptance criteria allows the target function to deteriorate within a finite range, and it is decided by a parameter similar to the temperature control parameter in physical process. Combined with genetic algorithm, the acceptance criteria can prevent premature convergence.

**Termination.** The algorithm terminates when the global optimal solution does not change in a continuous  $K$ -generation and when the per-set maximum number of iterations is completed.

#### 3.2.3. Steps of the Improved Genetic Algorithm

**Step 1.** The GA parameters are initialized. Population size is `pop_size`, the largest breeding algebra is `max_gen`, the number of subpopulations is  $n$ , the cross rate and variation rate of sub-population are  $F_{ic}$  and  $F_{im}$ , respectively, the global optimal solution unchanged termination of algebraic  $K$ , the subpopulation independent evolution of the  $Z$  algebra, global evolution of algebra `loop1 = 1`, general sub-population evolutionary algebra `loop2 = 1`, the initial annealing temperature is  $T$ , the minimum temperature is  $t$ , and the annealing coefficients are  $c$ .

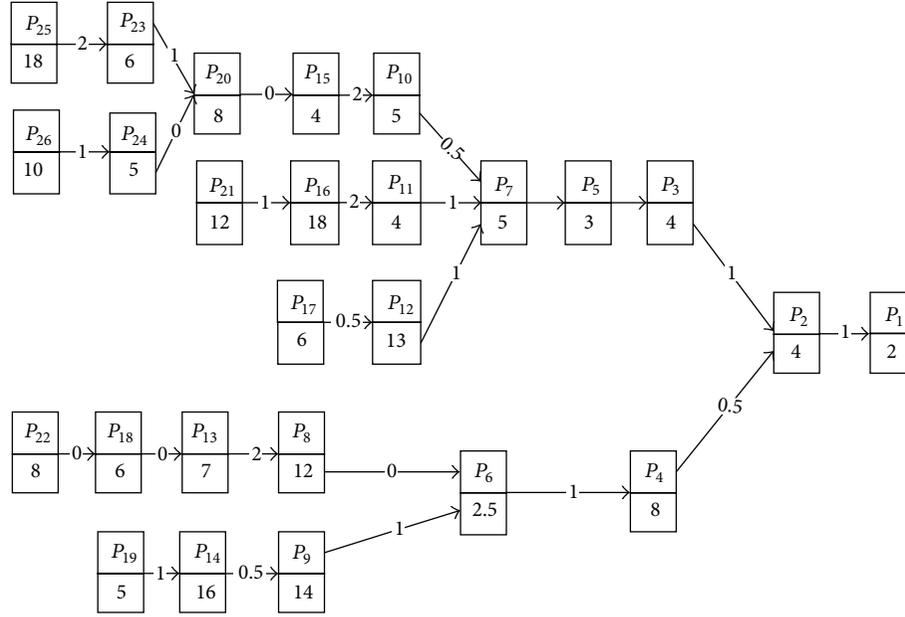


FIGURE 4: Timing constraints of tasks.

Step 2. The following operations are implemented for each general sub-population  $i$ .

- (a) The sub-populations in the fitness of each individual are evaluated.
- (b)  $x_{ij}$ ,  $x_{ik}$  are randomly selected from the sub-population. Two new individuals  $x'_{ij}$ ,  $x'_{ik}$  are crossed over and generated, and their fitness values are  $f(x'_{ij})$  and  $f(x'_{ik})$ . If  $f(x'_{ij}) \leq f(x_{ij})$ , accept  $x_{ij}$ ; if  $\min \{1, \exp(-(-f(x'_{ij}) - f(x_{ij}))/T)\} > \text{random}$  and  $f(x'_{ij}) > f(x_{ij})$ , accept  $x'_{ij}$ . The acceptance of  $x_{ik}$  and  $x'_{ik}$  are the same as before.

Step 3. Mutation. Whether the new individual is accepted according to Step 2 is determined.

Step 4. If  $\text{loop1} < Z$ ,  $\text{loop1} = \text{loop} + 1$ , Step 2 is repeated; otherwise, Step 5 is followed.

Step 5. The best individual of each of the general population (20%) is used to establish an excellent population. Crossover and mutation according to Steps 2 and 3 in the excellent population are performed.

Step 6. Form a mixed population by several general populations, and replace the individuals of the mixed population by of mixed population. Excellent populations before 20% individual replacement of the mixed population are then divided into new general populations.

Step 7. If the optimal solution is the same algebra  $K$ , this solution is obtained, and the algorithm terminates; otherwise, Step 8 is followed.

Step 8. If  $\text{loop2} < \text{max\_gen}$ ,  $T = cT$ ,  $c \in (0, 1)$ , and  $\text{loop2} = \text{loop2} + 1$ , Step 2 is repeated; otherwise, the algorithm is terminated.

## 4. Experiment

4.1. *Examples of Production Scheduling.* Enterprise  $E_1$  could not independently complete a device manufacturing task. It divides the task into 26 subtasks ( $P_1 \dots P_{26}$ ). Finally,  $E_1$  establishes a VE with four partners ( $E_2$ ,  $E_3$ ,  $E_4$ , and  $E_5$ ) through a tender:  $E_1$  implementation of the task set  $\{P_1, P_5, P_7, P_{14}, P_{23}, P_{26}\}$ ,  $E_2$  implementation of the task set  $\{P_4, P_9, P_{15}, P_{20}, P_{24}\}$ ,  $E_3$  implementation of the task set  $\{P_2, P_{10}, P_{12}, P_{16}, P_{19}\}$ ,  $E_4$  implementation of the task set  $\{P_6, P_8, P_{11}, P_{17}, P_{25}\}$ , and  $E_5$  implementation of the task set  $\{P_3, P_{13}, P_{18}, P_{21}, P_{22}\}$ . The task execution order constraints are shown in Figure 4. The top of the rectangular box represents the task number, the bottom represents the mission time, and the arrow line numbers indicate the transit time (when the production tasks are completed by the same enterprise, the transit time is 0).

4.2. *Scheduling Optimization.* All parameters are initialized, and MATLAB programming is used to achieve the improved multipopulation GA. The initial population size is 100, which is then divided into three general populations (with sizes of 30, 30, and 40). The crossover and mutation probabilities of the three populations are 0.8, 0.1; 0.85, 0.15; and 0.9, 0.2, respectively. The length of the chromosome is 26. The sub-population evolution algebra is 10, and the maximum evolution generation of the GA is 100. The initial annealing temperature, annealing coefficient, and lowest temperature are  $100^\circ\text{C}$ , 0.97, and  $10^\circ\text{C}$ , respectively. After implementation

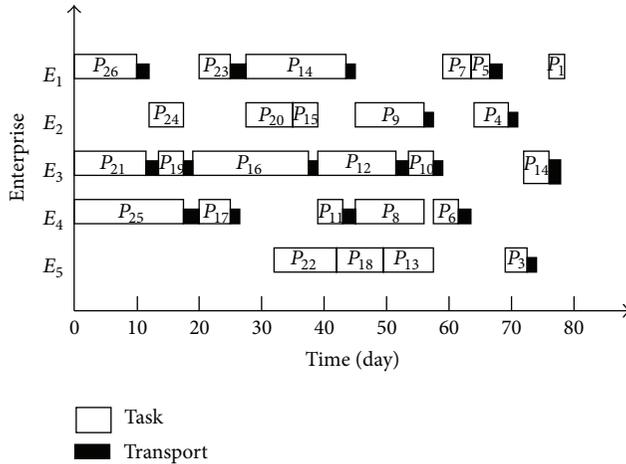


FIGURE 5: Optimal plan program by IMGSA.

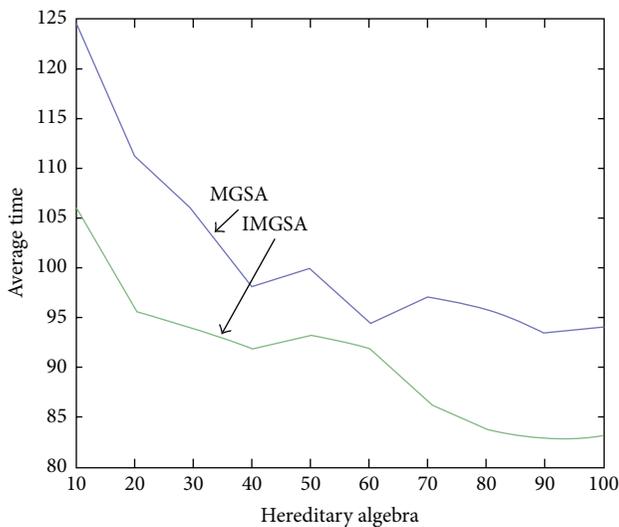


FIGURE 6: The average time comparison chart of MSGA and IMGAS.

of the algorithm, the optimal manufacturing cycle is 77.5 days. The optimal scheme using the Gantt chart is shown in Figure 5.

Using the above algorithm parameters, the unimproved multi-population (UMGSA) and improved multi-population algorithm (IMGSA) are run for 100 generations. The average completion time of the task with the convergence of genetic algebraic diagram is shown in Figure 6. The improved GA convergence rate is fast.

## 5. Conclusion

This paper presents a model for solving production planning. The improved GA is used to solve the model. It provides a good method for solving the problem of the production planning of VE. The convergence rate of the improved GA is faster than that of the unimproved GA. Thus, an optimal production plan is obtained quickly to guide the production of VE. During the production operation, the production plan

may change any time because companies are subjected to certain external factors. Therefore, the unfinished task of concentration can be based on the information of the completed task, and this algorithm can be used for adjustments.

## Acknowledgments

This work was supported by some grants from Guangdong Province Production Education and Scientific Study Programs, China (no. 2012B091100490 and no. 2011B090400056) and the Fundamental Research Funds for the Central Universities, SCUT.

## References

- [1] H. Helaakoski, P. Iskanius, and I. Peltomaa, "Agent-based architecture for virtual enterprises to support agility," *IFIP International Federation for Information Processing*, vol. 243, pp. 299–306, 2007.
- [2] A. Goel, H. Schmidt, and D. Gilbert, "Towards formalizing virtual enterprise architecture," in *Proceedings of the 13th Enterprise Distributed Object Computing Conference Workshops (EDOCW '09)*, pp. 238–242, September 2009.
- [3] N. Capuano, S. Miranda, F. Orcioli, and S. Vassallo, "E-Learning at work in the knowledge virtual enterprise," in *Proceedings of the 2nd International Conference on Complex, Intelligent and Software Intensive Systems (CISIS '08)*, pp. 507–512, March 2008.
- [4] M. H. Danesh, B. Raahemi, and M. A. Kamali, "A framework for process management in service oriented virtual organizations," in *Proceedings of the 7th International Conference on Next Generation Web Services Practices (NWeSP '11)*, pp. 12–17, October 2011.
- [5] S. Esparcia, R. Centeno, R. Hermoso, and E. Argente, "Artifacting and regulating the environment of a virtual organization," in *Proceedings of the 23rd IEEE International Conference on Tools with Artificial Intelligence (ICTAI '11)*, pp. 547–554, November 2011.
- [6] J. A. Crispim and J. P. De Sousa, "Partner selection in virtual enterprises," *International Journal of Production Research*, vol. 48, no. 3, pp. 683–707, 2010.
- [7] T. Jarimo and A. Salo, "Multicriteria partner selection in virtual organizations with transportation costs and other network interdependencies," *IEEE Transactions on Systems, Man and Cybernetics C*, vol. 39, no. 1, pp. 124–129, 2009.
- [8] N. Nayak, K. Prasanna, S. Datta, S. S. Mahapatra, and S. Sahu, "A novel swarm optimization technique for partner selection in virtual enterprise," in *IEEE International Conference on Industrial Engineering and Engineering Management (IEEM '10)*, pp. 1118–1122, December 2010.
- [9] D. Simona and P. Raluca, "Intellegent modeling method based on genetic algorithm for partner selection in virtual organizations," *Business and Economic Horizons*, vol. 5, no. 2, pp. 23–24, 2011.
- [10] A. M. Mohamed, "Optimal composition of virtual enterprises with interval cost parameters," in *Proceedings of the 8th International Conference on Informatics and Systems (INFOS '12)*, pp. BIO-188–BIO-194, 2012.
- [11] N. Yalaoui, L. Amodeo, F. Yalaoui, and H. Mahdi, "Particle swarm optimization under fuzzy logic controller for solving a hybrid reentrant flow shop problem," in *Proceedings of the IEEE*

- International Symposium on Parallel and Distributed Processing, Workshops and Phd Forum (IPDPSW '10)*, pp. 1–6, April 2010.
- [12] F. Dugardin, L. Amodeo, and F. Yalaoui, “Multiobjective scheduling of a reentrant hybrid flowshop,” in *Proceedings of the 39th International Conference on Computers and Industrial Engineering (CIE '09)*, pp. 193–198, July 2009.
- [13] Y. Gao and Z. B. Jiang, “Hybrid genetic algorithm for virtual production planning,” *Control and Decision-Making*, vol. 22, no. 8, 2007.
- [14] Y. Ding, Y. Gao, and G. Luo, “Virtual enterprise global production planning based on cost optimization with time constraints,” in *Proceedings of the International Conference on Management and Service Science (MASS '09)*, September 2009.
- [15] Q. J. Song, *The Cloud of Virtual Enterprise Production Planning and Control*, Silicon Valley, 2008.
- [16] Y. Gao and Y. S. Ding, *Virtual Enterprise Production Planning Model*, vol. 22, Enterprise Management, 2009.
- [17] Q. S. Li and J. G. Liu, “Research on VE production planning based on multi-agent,” *Research and Design*, vol. 22, no. 1, 2008.
- [18] L. M. Camarinha-Matos and H. Afsarmanesh, *VE Modeling and Support Infrastructures: Applying Multi-Agent System Approaches*, Springer, New York, NY, USA, 2001.
- [19] Q. Zhao and M. Zhou, “Virtual production tasks based on ant colony optimization scheduling,” *Wuhan University of Technology*, vol. 34, no. 3, 2011.
- [20] Z. Tao and L. Y. Xie, “Based on hybrid genetic algorithm for job shop scheduling problem,” *Computer Engineering and Applications*, vol. 18, 2005.
- [21] M. Huang, X. Wang, F.-Q. Lu, and H. Bi, “A coordination of risk management for supply chains organized as virtual enterprises,” *Mathematical Problems in Engineering*, vol. 2013, Article ID 931690, 11 pages, 2013.
- [22] X. Y. Guo and L. Jin, “The gray relational MCDM TOPSIS,” *Technology and Management*, vol. 12, no. 5, 2010.
- [23] J. Singh and A. Madhukar, “New method for calculating non-ideal point defect induced electronic structure,” *Solid State Communications*, vol. 41, no. 12, pp. 947–950, 1982.
- [24] W. Zhou and Y. Bu, “Cultural algorithm based on particle swarm optimization for partner selection of virtual enterprise,” in *Proceedings of the 31st Chinese Control Conference*, 2012.
- [25] S.-L. Yang, H.-W. Kang, and H. Zhou, “Research on web service-based virtual enterprise integration framework,” in *Proceedings of the 3rd International Conference on System Science, Engineering Design and Manufacturing Informatization*, 2012.
- [26] M. A. O. Pessoa, F. Junqueira, and D. J. S. Filho, “Virtual enterprise planning system using time windows and capacity constraint concepts,” in *Proceedings of the 38th Annual Conference on IEEE Industrial Electronics Society (IECON '12)*, pp. 2851–22856, 2012.
- [27] C. F. M. Toledo and J. M. G. Lima, “A multi-population genetic algorithm approach for PID controller auto-tuning,” in *Proceedings of the IEEE 17th Conference on Emerging Technologies & Factory Automation (ETFA '12)*, pp. 1–8, 2012.
- [28] B. Xiaojun and L. Guangxin, “The improvement of ant colony algorithm based on the inver-over operator,” in *Proceedings of the IEEE International Conference on Mechatronics and Automation (ICMA '07)*, pp. 2383–2387, August 2007.
- [29] Q. Xu, J. Mao, and Z. Jin, “Simulated annealing-based ant colony algorithm for tugboat scheduling optimization,” *Mathematical Problems in Engineering*, vol. 2012, Article ID 246978, 22 pages, 2012.

## Research Article

# Semisupervised Clustering for Networks Based on Fast Affinity Propagation

Mu Zhu, Fanrong Meng, and Yong Zhou

*Department of Computer Science and Technology, China University of Mining and Technology, Xuzhou 221008, China*

Correspondence should be addressed to Mu Zhu; [zhumubest@163.com](mailto:zhumubest@163.com)

Received 3 May 2013; Accepted 1 July 2013

Academic Editor: Orwa Jaber Housheya

Copyright © 2013 Mu Zhu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Most of the existing clustering algorithms for networks are unsupervised, which cannot help improve the clustering quality by utilizing a small number of prior knowledge. We propose a semisupervised clustering algorithm for networks based on fast affinity propagation (SCAN-FAP), which is essentially a kind of similarity metric learning method. Firstly, we define a new constraint similarity measure integrating the structural information and the pairwise constraints, which reflects the effective similarities between nodes in networks. Then, taking the constraint similarities as input, we propose a fast affinity propagation algorithm which keeps the advantages of the original affinity propagation algorithm while increasing the time efficiency by passing only the messages between certain nodes. Finally, by extensive experimental studies, we demonstrate that the proposed algorithm can take fully advantage of the prior knowledge and improve the clustering quality significantly. Furthermore, our algorithm has a superior performance to some of the state-of-art approaches.

## 1. Introduction

Network as an expressive data structure is popularly used to model structural relationships between objects in many real-world applications. Examples include social networks, web, and citation networks. In these networks, individuals represented by nodes are linked with some special relationships, such as the friendships between people, hyperlinks between web pages, and references among papers.

Network clustering, as a key technology for network analysis, can discover the hidden structures and functions in networks [1], which is attracting a considerable amount of attention from researchers in various domains. A lot of related research achievements, including modularity optimization [2–5], spectral clustering [6–8], and similarities-based algorithms [9–11] have been implemented to partition the networks into clusters (or communities, groups), such that there are a dense set of edges within each cluster and few edges between clusters. However, most of the existing algorithms are unsupervised and cannot utilize any prior knowledge to improve the clustering quality. Usually, prior knowledge related to the data such as label information or

pairwise constraints can be obtained in many real applications.

Semisupervised clustering is a very useful strategy that can guide the clustering process to get a high clustering quality in the presence of the obtained prior knowledge. There have been a lot of semisupervised clustering methods in data mining and machine learning domains, such as the constraint-based methods [12–14], the distance (or similarity) metric learning methods [15–18], and the methods integrating the two above [19–21]. However, most of them are designed for the traditional vector data and not well fit for the current network data. For example, some distance-based algorithms use the Euclidean distance, which is invalid in measuring the distances between nodes in networks. Only a few graph-based semisupervised methods, such as semisupervised spectral clustering algorithm [22–24] and semisupervised kernel  $k$ -means algorithms [25, 26], can be applied in network clustering, but they cannot always get clustering result with high and steady quality.

In this paper, we propose a novel semisupervised clustering algorithm for networks based on fast affinity propagation (SCAN-FAP), which has a strong ability to improve the

clustering quality by making full use of a little of prior knowledge. The basic innovation of our approach consists of two important components. The first is to define an effective similarity measure taking into account both the structural and the pairwise constraint information, while the second is to provide a performable clustering algorithm taking the similarities as input. Our contributions are threefold.

- (i) We propose a constraint similarity measuring method, which is extending the idea of the basic SimRank [27]. This method adopts the most commonly used must-link and cannot-link pairwise constraints as the prior knowledge and embodies them into the SimRank equation, which tactfully integrates the structural information and the prior knowledge.
- (ii) Taking the constraint similarities as input, we propose a fast affinity propagation algorithm for network clustering. By the analysis of the factor graph, the theoretical basis of the affinity propagation, we modify the binary model by reducing the number of variables to be optimized. The proposed algorithm derived from the new model can generate clusters with high quality while speed up the running time compared with the original affinity propagation algorithm [28].
- (iii) We demonstrate, by using various real networks, that the proposed algorithm method has the advantages of both the effectiveness and efficiency on improving the clustering quality, by making use of a small number of pairwise constraint information.

The paper is organized as follows. Section 2 reviews some related work including network clustering and semisupervised clustering. Section 3 describes the proposed constraint similarities and its calculations. Section 4 proposes the fast affinity propagation algorithm for networks. Section 5 analyzes the time complexity of our algorithm. We present the experimental results to show the advantages of our algorithm in Section 6. Finally, in Section 7, we present the conclusions of our work.

## 2. Related Work

**2.1. Network Clustering.** Recently, there exists an increasing body of literature on network clustering methods, such as modularity optimization, spectral clustering, and similarity-based algorithms.

Modularity [2] is a very famous objective function for measuring the strength of dividing a network into clusters. The value of modularity is the fraction of the edges that fall within the given clusters minus the expected such fraction if edges were distributed at random. Networks with high modularity have dense links between nodes within clusters but sparse connections between nodes in different clusters. Therefore, lots of optimization methods, such as Mod-CSA [3], LPAm [4], and MIGA [5], are designed for getting the maximum modularity as their objective. It is worth mentioning that these methods have a resolution limit and may fail to identify some smaller clusters [29].

Spectral clustering algorithms are also popularly used for networks. They cut the networks by transforming the initial set of nodes into a set of points in space, whose coordinates are elements of eigenvectors. The set of points is then clustered via standard techniques, such as  $k$ -means. There are various kinds of spectral clustering, which are distinguished from each other by their “cut functions,” which will be then solved as quadratic optimization problems. The traditional objective functions include the famous “Ratio Cut” [7] and “Normalized Cut” [8]. Other recent “cut functions” proposed in [30, 31] are set equal to the modularity for real networks. In addition, Dhillon et al. [32] proved that, in certain conditions, spectral clustering algorithms can seem as the corresponding kernel-based  $k$ -means formations.

A nature property in networks is the similarity between nodes. Therefore, some similarity-based algorithms are proposed to utilize the similarities to partition the networks. SCAN [9], a density-based algorithm, clusters the networks by using the transitivity of the cosine similarities between nodes. It can not only find the clusters with connectivity and maximality but also detect the hubs and outliers. Liu [10] made use of the famous affinity propagation algorithm to partition the networks with various similarity measures. Cheng et al. [11] measured the similarities between nodes by considering the information of both the content and links and then proposed an efficient incremental computing approach for network clustering.

However, most of algorithms introduced above are unsupervised, which do not have the ability of utilizing a small number of prior knowledge to improve the clustering quality.

**2.2. Semisupervised Clustering.** Semisupervised clustering is a kind of learning methods, which can guide the clustering process by using a small number of prior knowledge. It becomes growing popular in machine learning and data mining due to its important ability of improving the clustering quality. Generally, there are two kinds of prior knowledge: label information and pairwise constraints. The pairs of constraints are more popular than the label information since they are faster or cheaper to be obtained [19]. Must-link and cannot-link are two common used pairwise constraints. The former indicates the two individuals must be in the same cluster while the latter indicates the individuals must be in different clusters.

There are two kinds of strategies about semisupervised clustering: constraint-based optimization and distance (or similarity) metric learning. The constraint-based optimization methods, such as [12–14], modify the objective function by using the obtained label information or pairwise constraints and then solve the new objective optimization problem. The metric learning methods always use the prior information to adjust the original distances (or similarities) between objects and then cluster the dataset by some original method. Relevant examples are demonstrated in [15–18]. Some other methods, such as [19–21], combine the two strategies together to make use of the prior knowledge.

However, most of the existing semisupervised clustering methods will be invalid in network clustering, since they are

only designed for the traditional vector data. A few graph-based methods in their semisupervised formation can be used to partition networks. For example, semisupervised spectral clustering [22] modifies the adjacent matrix of the network using the pairwise constraints at the beginning of the original algorithm; semisupervised kernel-based  $k$ -means method proposed by Kulis et al. [25] defines a penalized kernel matrix subject to pairwise constraints, which will guide the assignments of the clusters; the method in [26] defines a new penalized kernel matrix taking the density modularity [33] as the objective function.

### 3. Constraint Similarity Measure

One of the most important motivations of our method is to adjust the original similarities between nodes when some prior knowledge consists in the networks. For the prior knowledge, we adopt the pairwise constraints since it is more popular in real applications. For the similarities adjustments, we define a novel measure integrating the pairwise constraints and the basic SimRank similarities.

**3.1. Basic SimRank.** SimRank is a general and a global similarity measure, which takes fully into account the structural context of all the nodes in networks. The basic idea of SimRank is “two objects are similar if they are related to similar objects.”

Let  $G = (V, E)$  represent a network, where  $V$  is the set of nodes and  $E$  is the set of edges. For a node  $v$ ,  $I(v) = \{u \mid (u, v) \in E\}$  represents the set of neighbors of  $v$ ;  $|I(v)|$  is the number of elements in  $I(v)$ ;  $I_i(v)$  is the  $i$ th element in  $I(v)$ . Let us denote the similarity between nodes  $a$  and  $b$  by  $s(a, b) \in [0, 1]$ . Following the earlier definitions, a recursive equation is defined for  $s(a, b)$ . If  $a = b$ , then  $s(a, b)$  is set to be 1.

Otherwise,

$$s(a, b) = \frac{C}{|I(a)||I(b)|} \sum_{i=1}^{|I(a)|} \sum_{j=1}^{|I(b)|} s(I_i(a), I_j(b)), \quad (1)$$

where  $C$  is a decay factor between 0 and 1. Note that either  $a$  or  $b$  may have no neighbors. In this situation,  $s(a, b)$  is set to be 0 since there is no way to infer any similarity between them. Formally, when  $I(a) = \emptyset$  or  $I(b) = \emptyset$ ,  $s(a, b) = 0$ .

**3.2. Definition of the Constraint Similarity Measure.** The basic SimRank is an unsupervised measure. However, in semisupervised network clustering, the prior knowledge should be taken fully into account for the similarity measure. The pairwise constraints are adopted to express the prior knowledge in our method. The formation of the pairwise constraints is a constraint matrix  $Q$ , the element of which represents the relationship between two nodes:

$$Q(i, j) = \begin{cases} 1 & \text{must-link } (i, j) \\ -1 & \text{cannot-link } (i, j) \\ 0 & \text{otherwise,} \end{cases} \quad (2)$$

where must-link means the pairs of two nodes must be in the same cluster and cannot-link means the pairs of two nodes must be in different clusters.

In particular to deserve to be mentioned, the must-link relationship satisfies three properties: reflexivity, symmetry, and transitivity. Formally,

- (1) reflexivity: for any node  $v$ ,  $Q(v, v) = 1$ ;
- (2) symmetry: for any pair of two nodes  $v$  and  $w$ , if  $Q(v, w) = 1$ , then  $Q(w, v) = 1$ ;
- (3) transitivity: for any three nodes  $u$ ,  $v$ , and  $w$ , if  $Q(u, v) = 1$  and  $Q(v, w) = 1$ , then  $Q(u, w) = 1$ .

Extending the basic SimRank and combining with the known information of the pairs of constraints, we define a novel constraint similarity measure composed of three composed three components as follows:

$$s(a, b) = \begin{cases} 1 & \text{if } Q(a, b) = 1 \\ 0 & \text{if } Q(a, b) = -1 \\ \frac{C}{|I(a)||I(b)|} \sum_{i=1}^{|I(a)|} \sum_{j=1}^{|I(b)|} s(I_i(a), I_j(b)) & \text{if } Q(a, b) = 0. \end{cases} \quad (3)$$

In (3), the similarity between two nodes is set to 1 as the max value when the must-link relationship is satisfied and 0 as the min value when cannot-link relationship is satisfied. Note that any node has the must-link relationship with itself and any isolated node with no neighbors has the cannot-link relationship with all the remaining nodes. The similarity between two nonconstraint nodes, similar to the basic SimRank equation, is calculated according to the similarities among their neighbors. The score will be increasing or decreasing when the two non-constraint nodes are linked with more must-link or cannot-link relationships. As a result, the constraint similarity measure considering the prior constraints information can help to improve the clustering result efficiently.

**3.3. Calculation of the Constraint Similarity Measure.** A solution to the SimRank equations for a network can be reached by iteration to a fix value. Let  $G$  be a given network with  $n$  nodes. For each iteration  $k$ , we store the  $n^2$  elements  $R_k(*, *)$ , where  $R_k(a, b)$  implies the score of the similarity between  $a$  and  $b$  in the current iteration. We successively calculate  $R_{k+1}(*, *)$  using  $R_k(*, *)$ . Initially, each  $R_0(a, b)$  is defined to be

$$R_0(a, b) = \begin{cases} 1, & \text{if } Q(a, b) = 1 \\ 0, & \text{else.} \end{cases} \quad (4)$$

To calculate  $R_{k+1}(a, b)$  from  $R_k(a, b)$ , we use (5), which has the same form as the basic SimRank equation, when there is no constraint between  $a$  and  $b$ ; namely,  $Q(a, b) = 0$ . On each

iteration  $k+1$ ,  $R_{k+1}(a, b)$  keeps 1 when  $Q(a, b) = 1$  and 0 when  $Q(a, b) = -1$ :

$$R_{k+1}(a, b) = \frac{C}{|I(a)||I(b)|} \sum_{i=1}^{|I(a)|} \sum_{j=1}^{|I(b)|} R_k(I_i(a), I_j(b)). \quad (5)$$

As shown previously, the current score of the similarity between  $a$  and  $b$  is updated by the score from previous iteration. We argue that this calculation method has the property of convergence.

**Proposition 1.** *The similarity score calculated by the method introduced before will converge to the true value as the iteration  $k$  increases; namely,  $s(a, b) = \lim_{k \rightarrow \infty} R_k(a, b)$ .*

*Proof.* Certainly,  $s(a, b)$  is always keeping the score 1 and 0 when  $Q(a, b) = 1$  and  $Q(a, b) = -1$ , respectively. So, we only need to prove the convergence for  $s(a, b)$  when  $Q(a, b) = 0$ . On the iteration  $k = 1$ , for any pair of nodes  $a$  and  $b$ ,

$$\begin{aligned} R_1(a, b) - R_0(a, b) &= R_1(a, b) \\ &= \frac{C}{|I(a)||I(b)|} \sum_{i=1}^{|I(a)|} \sum_{j=1}^{|I(b)|} R_0(I_i(a), I_j(b)), \\ &\quad R_0(I_i(a), I_j(b)) \in \{0, 1\} \\ &\leq \frac{C}{|I(a)||I(b)|} \sum_{i=1}^{|I(a)|} \sum_{j=1}^{|I(b)|} |I(a)||I(b)| \\ &= C. \end{aligned} \quad (6)$$

When  $k > 1$ ,

$$\begin{aligned} R_{k+1}(a, b) - R_k(a, b) &= \frac{C}{|I(a)||I(b)|} \sum_{i=1}^{|I(a)|} \sum_{j=1}^{|I(b)|} R_k(I_i(a), I_j(b)) \\ &\quad - \frac{C}{|I(a)||I(b)|} \sum_{i=1}^{|I(a)|} \sum_{j=1}^{|I(b)|} R_{k-1}(I_i(a), I_j(b)) \\ &= \frac{C}{|I(a)||I(b)|} \sum_{i=1}^{|I(a)|} \sum_{j=1}^{|I(b)|} (R_k(I_i(a), I_j(b)) \\ &\quad - R_{k-1}(I_i(a), I_j(b))) \\ &= \left( \frac{C}{|I(a)||I(b)|} \right)^2 \sum_{i=1}^{|I(a)|} \sum_{j=1}^{|I(b)|} \sum_{i=1}^{|I(a)|} \sum_{j=1}^{|I(b)|} (R_{k-1}(I_i(a), I_j(b)) \\ &\quad - R_{k-2}(I_i(a), I_j(b))) \end{aligned}$$

$$\begin{aligned} &= \left( \frac{C}{|I(a)||I(b)|} \right)^k \sum_{i=1}^{|I(a)|} \sum_{j=1}^{|I(b)|} \cdots \sum_{i=1}^{|I(a)|} \sum_{j=1}^{|I(b)|} (R_1(I_i(a), I_j(b)) \\ &\quad - R_0(I_i(a), I_j(b))). \end{aligned} \quad (7)$$

For  $\forall i \in I(a), j \in I(b), i$  and  $j$  satisfy  $0 \leq R_1(I_i(a), I_j(b)) - R_0(I_i(a), I_j(b)) \leq C$ ; therefore,

$$\begin{aligned} 0 &\leq R_{k+1}(a, b) - R_k(a, b) \\ &\leq \left( \frac{C}{|I(a)||I(b)|} \right)^k (|I(a)||I(b)|)^k C^k = C^{k+1}. \end{aligned} \quad (8)$$

Since  $C \in (0, 1)$ ,  $\lim_{k \rightarrow \infty} R_{k+1}(a, b) - R_k(a, b) = 0$ . So, it follows from the previous proposition that  $R(a, b)$  will converge to  $s(a, b)$ .  $\square$

#### 4. Fast Affinity Propagation Algorithm for Network Clustering

After the similarities are calculated, the following work we should do is to design an efficient and effective clustering method making use of the similarities. Although there are a number of candidate algorithms to be chosen, we adopt the famous affinity propagation algorithm (AP) [28], which has the ability to get a stable and reliable clustering result. Furthermore, we observe that the efficiency of AP can be increased by ingenious design, due to the sparse similarities in networks. So, in this section, we propose a fast affinity propagation algorithm for network clustering.

**4.1. Factor Graph.** We start from the optimization problem of the factor graph, which is the theoretical basis of the affinity propagation algorithm. Figure 1 is the binary model for affinity propagation demonstrated in [34]. There are two types of nodes representing variables and functions, respectively, in the binary model. Let  $C = \{c_{11}, \dots, c_{mm}\}$  be the set of  $N^2$  binary variables in the graph, such that  $c_{ij} = 1$  if the exemplar for node  $i$  is node  $j$ . In this notion,  $c_{jj} = 1$  indicates that  $j$  is an exemplar itself. For each variable  $c_{ij}$ , there are a similarity function node and two constraint function nodes. The similarity function is written in (9), where the value is equal to the input similarity between nodes  $i$  and  $j$  if  $c_{ij} = 1$ . The 1-of- $N$  constraint function  $I$  denotes each node in affinity propagation clustering must be assigned to only a single exemplar. The consistency constraint function  $E$  means that if a node itself is not an exemplar, it cannot be the exemplar for any other node. The goal of the affinity propagation is to maximize  $S(c_{11}, \dots, c_{mm}) = \sum_{i,j} S_{ij}(c_{ij})$  by finding the optimal assignment for  $C$ . Obviously, the number of the variables to be solved is  $n^2$ :

$$S_{ij}(c_{ij}) = \begin{cases} s(i, j) & c_{ij} = 1, \\ 0 & c_{ij} = 0. \end{cases} \quad (9)$$

For semisupervised network clustering, the similarity between two nodes will be set to 0 when they are satisfied with

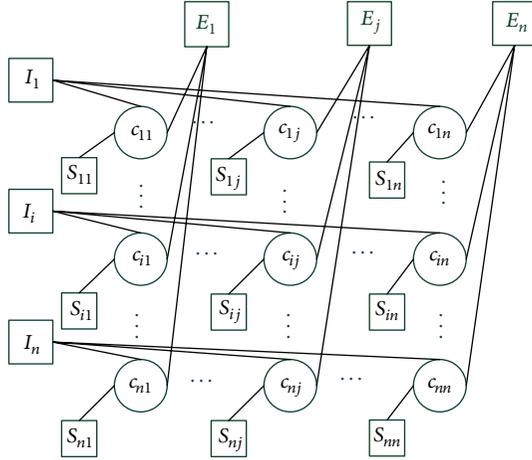


FIGURE 1: Binary model for factor graph.

cannot-link relationship. In the opposite way, there are lots of pairs of nodes with 0 similarity scores due to the influence both of the cannot-link relationships and the sparse structure of the real networks. Therefore, we argue that the pairs of nodes with 0 similarity scores are also constrained by cannot-link relationships. Formally, for nodes  $i$  and  $j$ ,  $s(i, j) = 0 \Leftrightarrow Q(i, j) = 0$ . Since the pairs of nodes must be assigned into different clusters, if the similarity score between  $i$  and  $j$  is 0, then  $i$  and  $j$  are not represented by each other as the exemplar, such that both  $c_{ij}$  and  $c_{ji}$  are set to 0. In this way, parts of the binary variables will be set to 0 in the optimum assignment of  $C$ , which makes the original factor graph in Figure 1 adjusted.

Firstly, to facilitate the description, we give the definition for the similarity neighborhoods. For a node  $v$ ,  $\Gamma(v)$  is defined as the similarity neighborhoods of  $v$  such that  $\Gamma(v) = \{w \mid s(v, w) \neq 0\}$ ; namely, the similarity score between  $v$  and any element in  $\Gamma(v)$  is greater than 0. In addition,  $\Gamma_k(v)$  denotes the  $k$ th item in  $\Gamma(v)$ , and  $K_v = |\Gamma(v)|$  denotes the number of the items in  $\Gamma(v)$ . Then, using the similarity neighborhoods we give the local factor graph for nodes  $i$  and  $j$  in Figures 2 and 3.

Figure 2 indicates the relationships between the function node  $I_i$  and the related binary variable nodes. If node  $j \notin \Gamma(i)$ , which means  $s(i, j) = 0$ , the corresponding  $c_{ij}$  is set to 0, so  $i$  cannot choose  $j$  as its exemplar. Otherwise, if  $j \in \Gamma(i)$ ,  $i$  is satisfied with the 1-of- $N$  constraint in (10), which means that  $i$  can only choose its exemplar in its similarity neighborhoods,

$$I_i(c_{ij}) = \begin{cases} -\infty & \text{if } \sum_{j \in \Gamma(i)} c_{ij} \neq 1 \\ 0 & \text{otherwise.} \end{cases} \quad (10)$$

Figure 3 indicates the relationships between the function nodes  $E_j$  and the related binary variable nodes. If node  $i \notin \Gamma(j)$ , which means  $s(i, j) = 0$ , the corresponding  $c_{ij}$  is set to 0. Otherwise, if  $i \in \Gamma(j)$ , then  $j$  is satisfied with the consistency constraint in (11), which means that all the nodes in  $\Gamma(j)$

cannot choose  $j$  as their exemplars if  $j$  is not an exemplar itself ( $c_{jj} = 0$ ),

$$E_j(c_{ij}) = \begin{cases} -\infty & \text{if } c_{jj} = 0, \sum_{i \in \Gamma(j)} c_{ij} > 0 \\ 0 & \text{otherwise.} \end{cases} \quad (11)$$

The whole factor graph is integrating all the function nodes  $I$  and  $E$  like the ones represented in Figures 2 and 3. Each variable node  $c_{ij}$  becomes an isolated node without any constraint, and its value is set to 0. As a result, the objective function of this new factor graph is defined to be

$$F(c_{ij} \mid s(i, j) \neq 0) = \sum_{i,j} S_{ij}(c_{ij} \mid s(i, j) \neq 0) + \sum_i I_i(c_{ij} \mid j \in \Gamma(i)) + \sum_j E_j(c_{ij} \mid i \in \Gamma(j)). \quad (12)$$

The problem of maximizing (12) can be solved by the max-sum algorithm [35], which passes messages between function nodes and variable nodes in the factor graph. As depicted in Figure 4, there are five message types passed between nonisolated variable nodes and function nodes. The corresponding message update rules are defined as (13) to (17), when executing the max-sum algorithm on the new factor graph,

$$\rho_{ij}(i \mid i \in \Gamma(j)) = s_{j \in \Gamma(i)}(i, j) + \eta_{j \in \Gamma(i)}(i, j), \quad (13)$$

$$\beta_{ij}(j \mid j \in \Gamma(i)) = s_{i \in \Gamma(j)}(i, j) + \alpha_{i \in \Gamma(j)}(i, j), \quad (14)$$

$$\eta_{ij}(j \mid j \in \Gamma(i)) = - \max_{k \neq j \& k \in \Gamma(i)} \beta_{ik}, \quad (15)$$

$$\alpha_{ij}(j \mid j \in \Gamma(i)) = \begin{cases} \sum_{k \neq j \& k \in \Gamma(j)} \max[\rho_{kj}, 0] & i = j \\ \min \left[ 0, \rho_{jj} + \sum_{k \notin \{i, j\} \& k \in \Gamma(j)} \max[\rho_{kj}, 0] \right] & i \neq j. \end{cases} \quad (16)$$

We eliminate  $\beta$  and  $\eta$ , while only keeping  $\alpha$  and  $\rho$ . As a result,  $\alpha$  keeps the formation as (16) being only dependent on  $\rho$ , while  $\rho$  is calculated by (17) being only dependent on  $\alpha$ ,

$$\rho_{ij}(i \mid i \in \Gamma(j)) = s(i, j) - \max_{k \neq j \& k \in \Gamma(i)} (s(i, k) + \alpha_{ik}). \quad (17)$$

Then, the optimal solution of the objective function can be obtained by updating  $\alpha$  and  $\rho$  alternately until the convergence. We can see that the number of messages passed is reduced on the factor graph due to the decrease of the elements of the variable nodes to be assigned. Therefore the time efficiency will be increasing for the objective optimizing.

**4.2. Clustering Algorithm.** As the analysis in the previous factor graph, we extend the original affinity propagation, an exemplar-based algorithm, into a fast version for network clustering. There are some basic parameters.

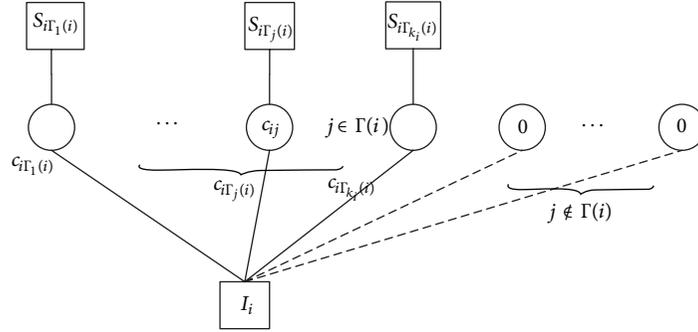


FIGURE 2: The relationships between function node  $I_i$  and the related binary variable nodes.

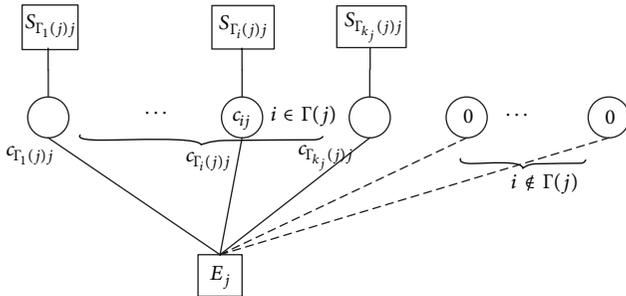


FIGURE 3: The relationships between function node  $E_j$  and the related binary variable nodes.

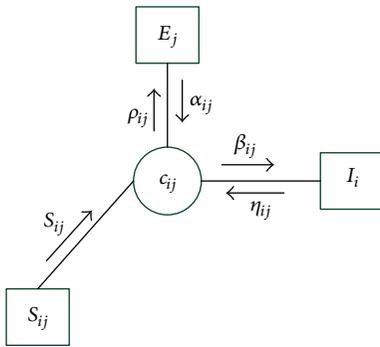


FIGURE 4: Messages passing between function nodes and variable nodes.

*Responsibility*  $r(i, j)$ , equivalent to  $\rho_{ij}$  in the factor graph, represents the evidence for how well-suited node  $j$  is to serve as the exemplar for node  $i$ .

*Availability*  $a(i, j)$ , equivalent to  $\alpha_{ij}$  in the factor graph, represents the evidence for how appropriate it would be for node  $i$  to choose node  $j$  as its exemplar.

*Preference*  $p$ , the initialized value  $s(i, i)$  for every node, represents the tendency of node  $i$  being chosen as an exemplar. At the beginning of the algorithm, all nodes share the same preference, which controls the number of the final clusters. The larger value of the preference results in a larger number of clusters.

*Damping factor*  $\lambda$  is used to smooth the responsibility and the availability so as to avoid numerical oscillations. In our algorithm,  $\lambda$  is set to 0.8. The smoothed responsibility and the availability are calculated by (18) and (19),

$$a(i, j | j \in \Gamma(i))^{(t)} = (1 - \lambda) a(i, j | j \in \Gamma(i))^{(t-1)} + \lambda a(i, j | j \in \Gamma(i))^{(t-1)}, \quad (18)$$

$$r(i, j | j \in \Gamma(i))^{(t)} = (1 - \lambda) r(i, j | j \in \Gamma(i))^{(t-1)} + \lambda r(i, j | j \in \Gamma(i))^{(t-1)}. \quad (19)$$

According to the parameters defined before, for a given network  $G = (V, E)$  and the constraint matrix  $Q$ , the basic process of our algorithm is as follows.

*Step 1.* Combined with the constraint matrix  $Q$ , calculate the similarities for all pairs of nodes in  $G$ .

*Step 2.* Construct the similarity neighborhoods for each node.

*Step 3.* Initialize the preference  $p$  and the maximum iteration number; for each node  $i$  and node  $j$  in the similarity neighborhoods of  $i$ , set  $r(i, j) = a(i, j) = 0$ .

*Step 4.* For each node  $i$  and node  $j$  in the similarity neighborhoods of  $i$ , update  $a(i, j)$  by (16) and (18); update  $r(i, j)$  by (17) and (19).

*Step 5.* For each node  $i$ , calculate its exemplar by (20):

$$c^* = \arg \max_{j \in \Gamma(i)} \{r(i, j) + a(i, j)\}. \quad (20)$$

*Step 6.* When the exemplars for all nodes are not changed any more, or the iteration reaches the maximum value, the algorithm will be terminated, and the nodes with the same exemplar will be partitioned into the same cluster. Otherwise, go to Step 4.

## 5. Complexity Analysis

In this section, we present an analysis of the computation complexity of the proposed algorithm. Given a network with  $n$  nodes, the algorithm is composed of two parts: the constraint similarities calculation and the fast affinity propagation clustering. We analyze the computation complexity of the two parts, respectively, in the following.

For the constraint similarities calculation, if there is no any constraint condition, the running cost is  $O(k_s d N^2)$ , where  $k_s$  is the number of iterations ( $k_s \ll N$ ) of the calculation and  $d$  is the average degree of all nodes in the network ( $d \ll N$  in many real networks). If some constraint information is obtained, the similarities between the pairs of nodes satisfied with constraint conditions need not to be calculated. This is because the value will be fixed to 1 and 0 for the must-link and cannot-link pairs all the time. Assuming the number of pairwise constraints is  $C$ , the total cost of the constraint similarities calculation is  $O(k_s d (N - C)^2)$ .

For the fast affinity propagation algorithm, the worst case is that the similarities between all pairs of nodes are greater than 0. In this situation, the time cost is  $O(k_A N^2)$ , where  $k_A$  is the number of iterations of the algorithm. However, the edges are sparse in many real networks, which results in the fact that the similarity scores are 0 between many pairs of nodes. The fast affinity propagation algorithm does not take into account the messages passing between the pairs of nodes provided with 0 similarity scores. As a result, let  $M$  ( $M < N^2$ ) represent the number of pairs of nodes provided with similarity scores greater than 0. Then, the total cost of the fast affinity propagation algorithm is  $O(k_A M)$ .

## 6. Evaluation

In this section, we evaluate the performance of the proposed semisupervised network clustering algorithm based on fast affinity propagation (SCAN-FAP). We conduct all the experiments on a Pentium Core2 Duo 2.8 GHz PC with 2 GBytes of main memory, running on Windows 7. We implement our algorithm in C++, using Microsoft Visual Studio 2008.

**6.1. Dataset.** We use six real network datasets to evaluate the performance of SCAN-FAP. Table 1 lists the information about these networks.

### 6.2. Effectiveness

**6.2.1. Evaluation Criteria.** Since the underlying class labels of all the datasets are already known, we adopt the Normalized Mutual Information (NMI) [25, 32] and the  $F$ -Measure Score [17] in our experiments to evaluate the quality of the clusters generated by various methods.

NMI is currently widely used in measuring the performance of network clustering algorithms. Formally, the measurement of NMI can be defined as follows:

$$\text{NMI} = \frac{-2 \sum_{i=1}^r \sum_{j=1}^k N_{ij} \log(N_{ij} N / N_i N_j)}{\sum_{i=1}^r N_i \log(N_i / n) + \sum_{j=1}^k N_j \log(N_j / N)}, \quad (21)$$

TABLE 1: Information about the experimental datasets.

Networks	Nodes	Edges	Clusters	References
Football	115	616	12	[36]
Polbooks	105	441	3	[36]
Adjnoun	112	425	2	[36]
PolBlogs	1490	19090	2	[36]
Cora	2708	5429	7	[37, 38]
Citeseer	3312	4732	6	[37, 38]

where  $N$  is the confusion matrix,  $N_{ij}$  is the number of nodes both in the  $i$ th class and the  $j$ th cluster,  $r$  and  $k$  are the number of classes and clusters, respectively, and  $N_i$  and  $N_j$  are the number of nodes in the  $i$ th class and the  $j$ th cluster, respectively.

$F$ -Measure Score is another commonly used measure for evaluating clustering algorithms. Assume  $T$  is the set of node pairs  $(i, j)$  where nodes  $i$  and  $j$  belong to the same classes in the ground truth and  $S$  is the set of node pairs that belong to the same clusters generated by an algorithm. Then the  $F$ -Measure Score is computed from both the *precision* and the *recall* synthetically:

$$F\text{-Measure} = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}}, \quad (22)$$

where *precision* and *recall* are written as follows (23):

$$\begin{aligned} \text{precision} &= \frac{|S \cap T|}{|S|}, \\ \text{recall} &= \frac{|S \cap T|}{|T|}. \end{aligned} \quad (23)$$

**6.2.2. Experimental Methods.** We compare SCAN-FAP with the other four representative semisupervised clustering algorithms, which are briefly described in the following.

**SS-SP**, a semisupervised spectral clustering algorithm [22], uses the known constraint relationships to modify the adjacent matrix of the original network. If the nodes  $i$  and  $j$  satisfy the must-link constraint,  $A_{ij}$  is set to 1; if  $i$  and  $j$  satisfy the cannot-link relationship,  $A_{ij}$  is set to 0. SS-SP will then execute the conventional spectral clustering algorithm on the modified adjacent matrix.

**SS-NCut-KK** [25] is a semisupervised kernel  $k$ -means algorithm based on Normalized Cut. In this algorithm, the original kernel matrix and the penalty term with constraint relationships are combined to construct a new kernel matrix, such that the wrong cluster assignments will go against the objective function. The kernel matrix is defined as  $K = \sigma I - L + W$ , where  $\sigma$  is a constant large enough to ensure that  $K$  is positive definite,  $I$  is the identity matrix,  $L$  is the corresponding Laplacian matrix, and  $W$  is the matrix of the penalty term. SS-NCut-KK executes the kernel  $k$ -means algorithm using the matrix  $K$ .

**SS-DM-KK** [26] is another semisupervised kernel  $k$ -means algorithm, which is similar to SS-NCut-KK. It takes

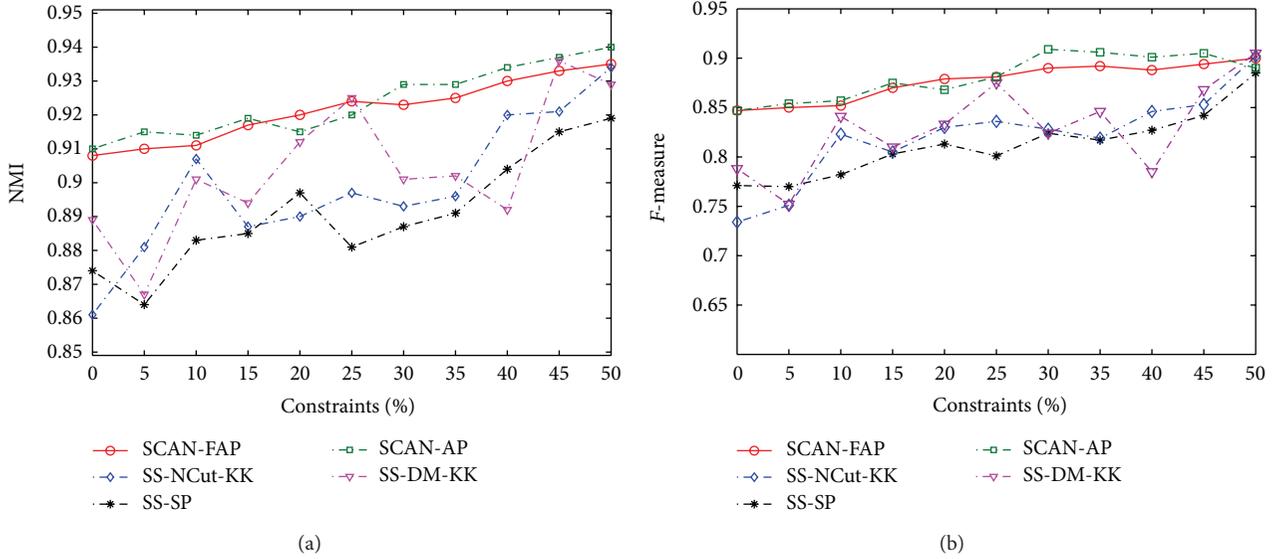


FIGURE 5: Clustering results on Football.

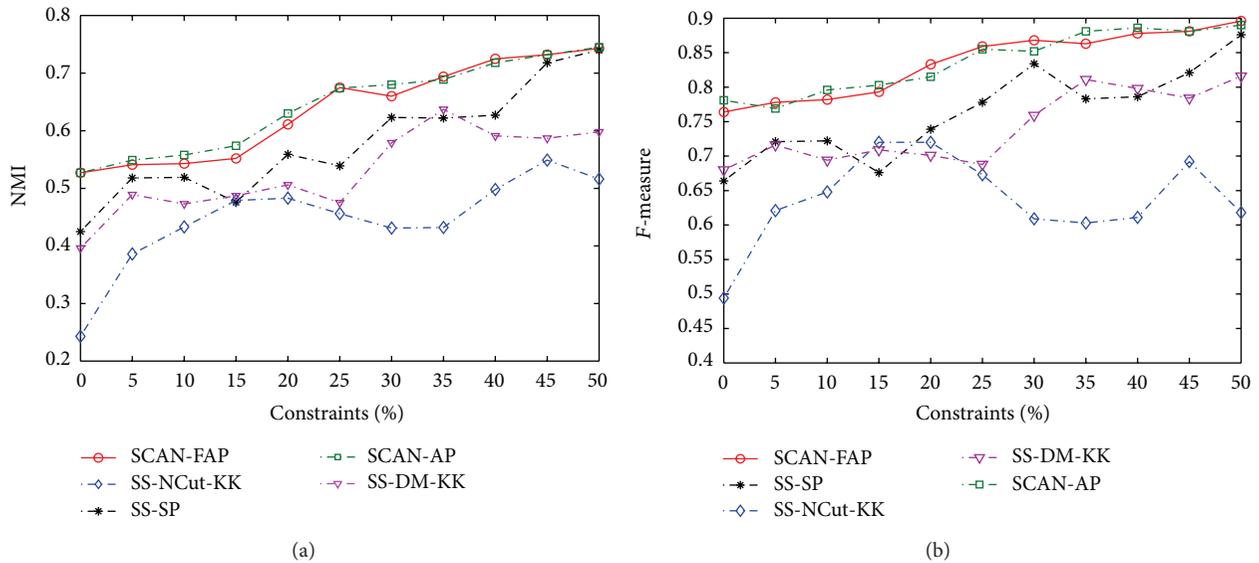


FIGURE 6: Clustering results on Polbooks.

the density modularity [33] integrating the constraint information as its objective function, which is the only difference from SS-NCut-KK.

SCAN-AP is a semisupervised network clustering algorithm most related to our work. The original affinity propagation algorithm is used taking as input the constraint similarities between all the nodes in the network.

We select pairwise constraints randomly. The proportion of the pairwise constraints to all pairs in the network is set to 5%, 10%, ..., 50%. Due to the randomness of the pairwise constraints, the clustering result of each experiment is the average value generated by running the corresponding algorithm over 20 times. The number of the clusters is set

to the number of real class labels in the networks. Since the number of clusters generated by SCAN-AP and SCAN-FAP depends on the value of the preference parameter  $p$ , we choose different value of  $p$  to adjust the number of clusters to the ground truth.

**6.2.3. Experimental Results.** Figures 5, 6, 7, 8, 9, and 10 demonstrate the curves of the clustering results on the six real datasets generated by different algorithms, where the horizontal coordinates represent the proportion of pairwise constraints and the vertical coordinates represent the NMI and  $F$ -Measure scores, respectively. We can get some observations as follows.

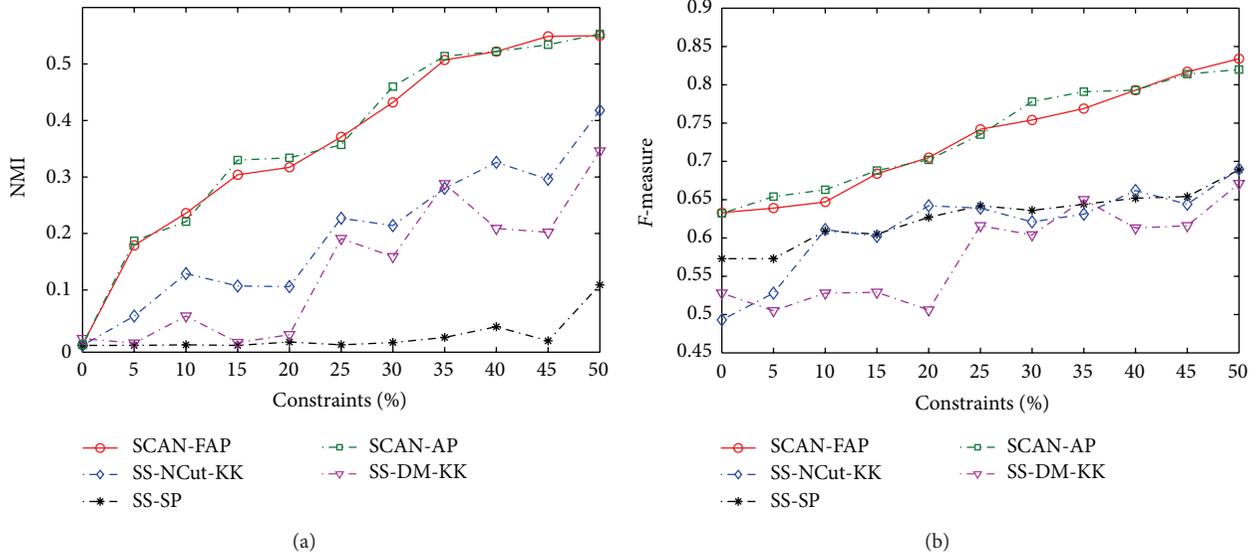


FIGURE 7: Clustering results on Adjnoun.

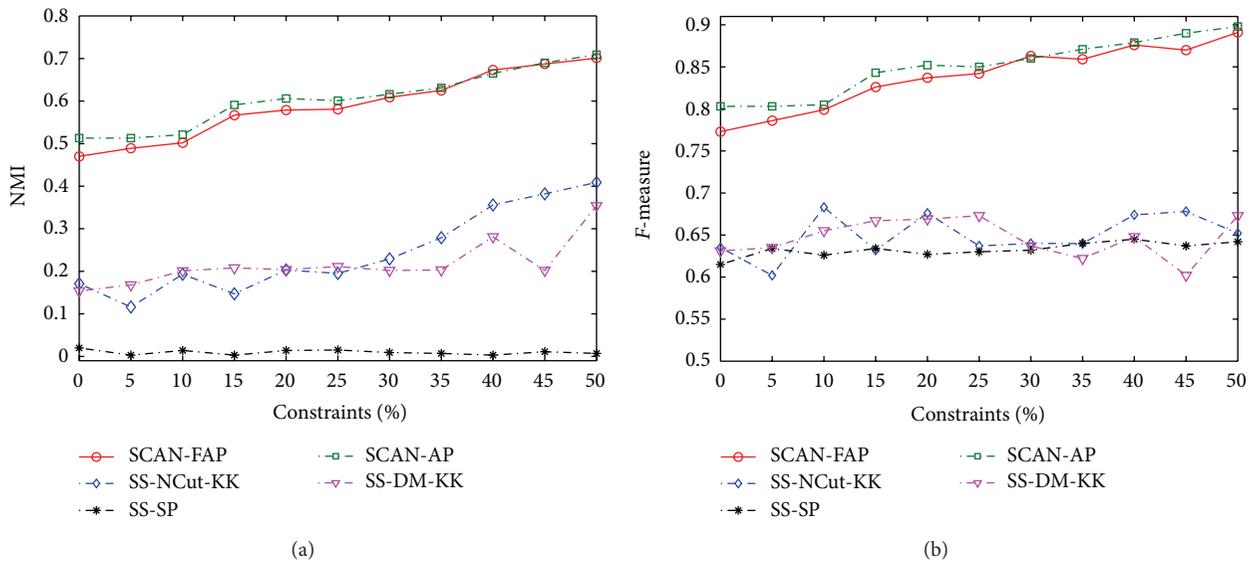


FIGURE 8: Clustering results on Polblogs.

Firstly, we compare all the algorithms in their unsupervised condition, which is reflected in the curves that the horizontal coordinates are 0. We can observe that SCAN-FAP and SCAN-AP get higher quality results on all the networks except Adjnoun than the other three algorithms. This indicates that SimRank is a good measure for similarities between nodes in networks. Note that the NMI of the results on Adjnoun obtained by all algorithms is close to 0, which implies that only the unsupervised structural information is insufficient for the clusters.

Then, we compare each semisupervised algorithm with its unsupervised version. When the pairwise constraints are provided, we can see from both NMI and *F*-Measure scores that the clustering results of SCAN-FAP and SCAN-AP are better

than the ones without any pairwise constraints provided. So, it can be considered that the proposed constraint similarity measure can take full advantage of the prior knowledge to guide the clustering process so as to improve the quality of the clustering results. For the other algorithms, SS-NCut-KK and SS-DM-KK can improve the quality of clustering in most cases, but SS-SP fails to utilize the pairwise constraints on Adjnoun and PolBlogs.

We also compare the performance of SCAN-FAP with other semisupervised clustering algorithms except SCAN-AP. The NMI and *F*-Measure curves of SCAN-FAP can display ascending trend together with increasing the pairwise constraints. However, the SS-SP, SS-NCut-KK, and SS-DM-KK show occasionally fluctuant or unascending states, which

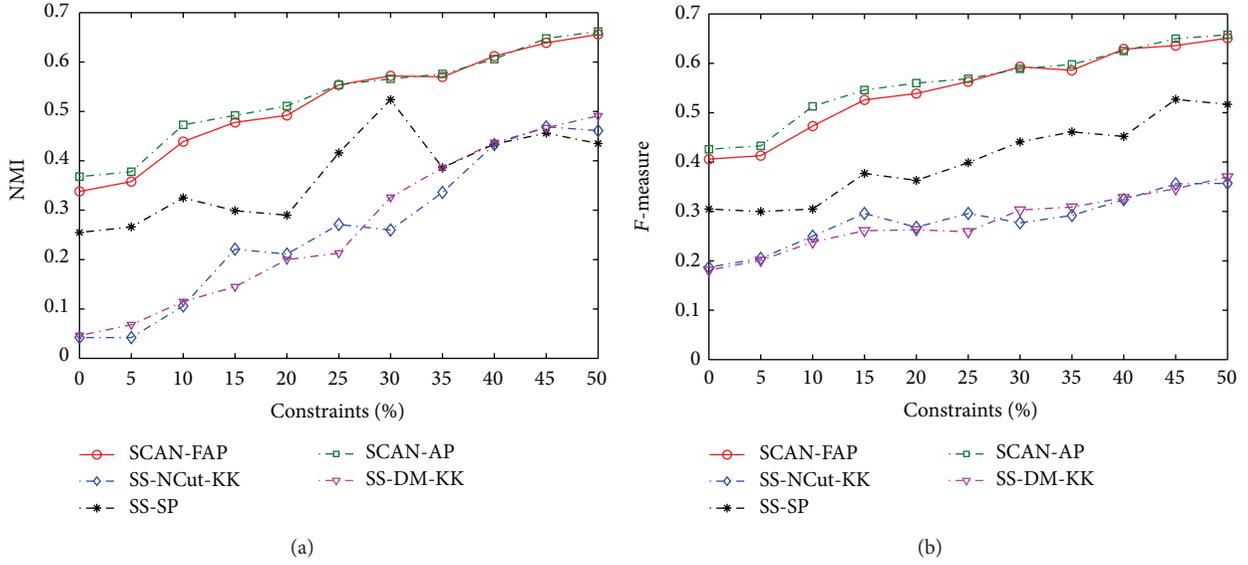


FIGURE 9: Clustering results on Cora.

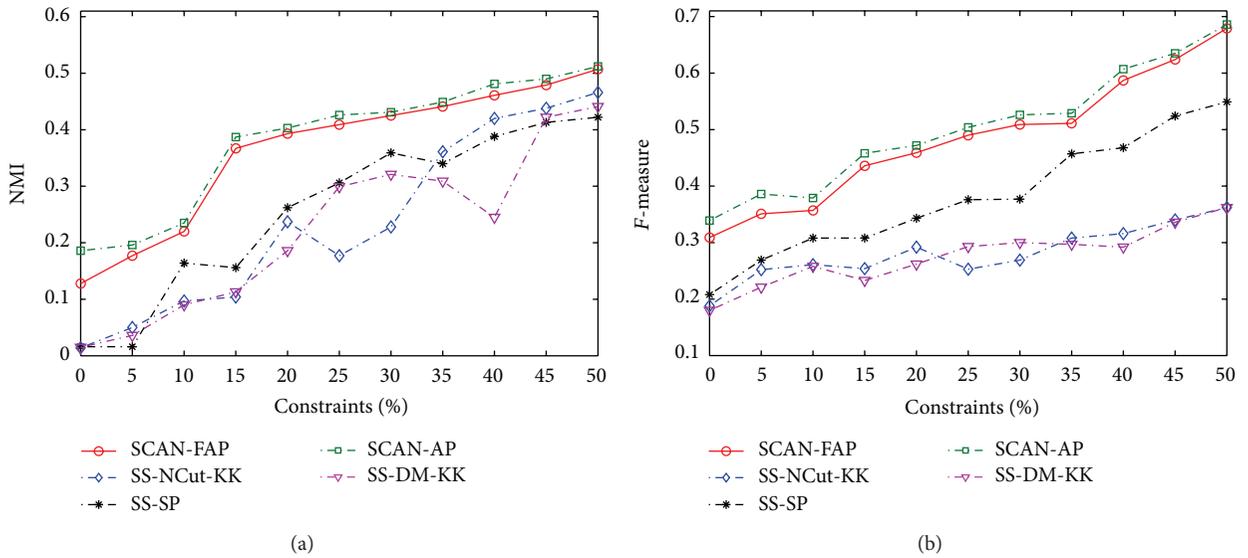


FIGURE 10: Clustering results on Citeseer.

reduces their reliability. For example, SS-SP on political blogs appears invalid with any percentage of constraints, while SS-NCut-KK and SS-DM-KK on Citeseer reveal decreased NMI with 25% and 40% constraints provided, respectively, and so on. One of the reasons for analyzing the instability of the three algorithms is their random initializations. For SS-SP, we should initialize the cluster centers when the  $k$ -means algorithm runs on the data composed of the eigenvectors and for SS-NCut-KK and SS-DM-KK, we should initialize the cluster assignment for each node.

Finally, we compare the performance of SCAN-FAP with its similar algorithm SCAN-AP. Meaningfully, the clustering quality of SCAN-FAP is always very close to SCAN-AP, which confirms the effectiveness of the messages passed only

in the pairs of nodes with similarity scores greater than 0. We demonstrate that SCAN-FAP can take the place of SCAN-AP under keeping the clustering quality. At the same time, the time efficiency is much increased, which will be demonstrated next.

**6.3. Efficiency.** From the effectiveness evaluations, we can see that SCAN-FAP and SCAN-AP have the ability of getting clusters with high quality. They only need to be executed one time in real applications due to the stability of them. However, the other three methods have to be run many times and average the results, since they are influenced by some factors such as the initializations. Therefore, we only compare the time efficiency of SCAN-FAP and SCAN-AP. In addition,

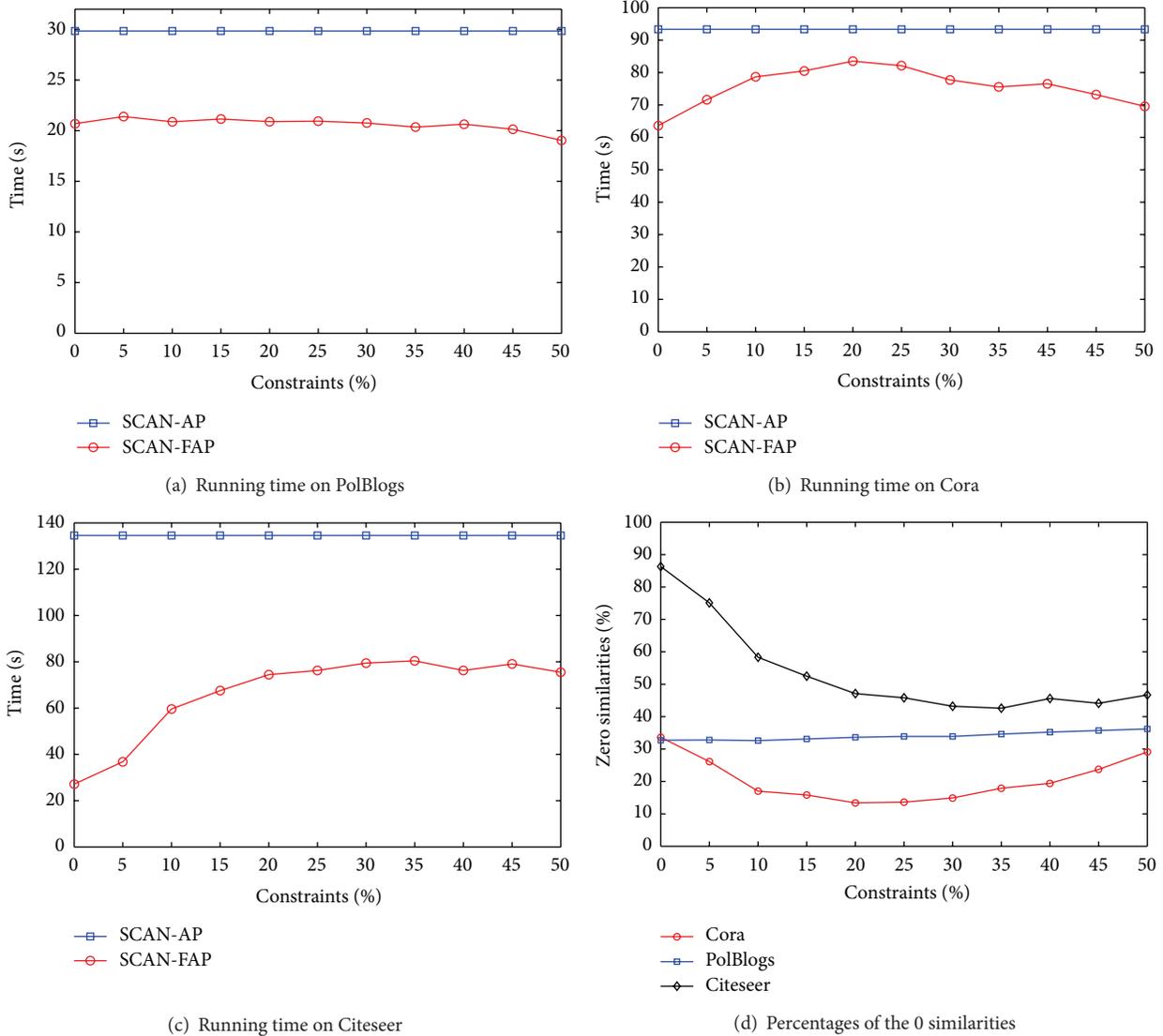


FIGURE 11: Efficiency comparison between SCAN-FAP and SCAN-AP.

because both the two algorithms share the same module that calculates the constraint similarities, we only take the running time of the clustering process for comparisons. The datasets are adopted by the three relatively large networks: PolBlogs, Cora, and Citeseer.

Figure 11 demonstrates the results about the efficiency comparisons, where (a), (b), and (c) are the running time of the two algorithms on three datasets, respectively, and (d) illustrates the proportion of the number of the pairwise nodes with 0 similarity scores to all the pairs in each network. We can observe that SCAN-AP keeps the running time consistently, since the messages are passed between all the pairs of nodes. SCAN-FAP can increase the time efficiency obviously, and the degree of the increasing is inversely proportional to the ratio of the pairs of nodes with 0 similarity scores. This is because each node only chooses the exemplar from its similarity neighborhoods when using fast affinity propagation algorithm in SCAN-FAP.

## 7. Conclusions

In this paper, we propose a novel semisupervised clustering algorithm for networks based on fast affinity propagation, which can utilize the prior knowledge to guide the unsupervised clustering process. The algorithm firstly defines a constraint similarity measure, which takes the pairwise constraints information to extend the basic SimRank measure. The similarity scores calculated with the new measure are influenced by the must-link and cannot-link relationships, such that the semisupervised property is achieved. Then, taking as input the constraint similarities, a fast affinity propagation algorithm is proposed to partition the networks. This algorithm maintains the advantages of the original affinity propagation algorithm. Moreover, it increases the running efficiency by passing messages between only the nodes with similarity scores greater than 0. Experimental studies, by various kinds of real networks, demonstrate that

our algorithm has a stronger ability of making use of a small number of prior knowledge than the other representative methods, to improve the clustering quality effectively and efficiently.

## References

- [1] M. Coscia, F. Giannotti, and D. Pedreschi, "A classification for community discovery methods in complex networks," *Statistical Analysis and Data Mining*, vol. 4, no. 5, pp. 512–546, 2011.
- [2] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Physical Review E*, vol. 69, no. 2, Article ID 026113, 2004.
- [3] J. Lee, S. P. Gross, and J. Lee, "Modularity optimization by conformational space annealing," *Physical Review E*, vol. 85, no. 5, Article ID 056702, 5 pages, 2012.
- [4] X. Liu and T. Murata, "Advanced modularity-specialized label propagation algorithm for detecting communities in networks," *Physica A*, vol. 389, no. 7, pp. 1493–1500, 2010.
- [5] R. Shang, J. Bai, L. Jiao, and C. Jin, "Community detection based on modularity and an improved genetic algorithm," *Physica A*, vol. 392, no. 5, pp. 1215–1231, 2013.
- [6] U. von Luxburg, "A tutorial on spectral clustering," *Statistics and Computing*, vol. 17, no. 4, pp. 395–416, 2007.
- [7] Y.-C. Wei and C.-K. Cheng, "Ratio cut partitioning for hierarchical designs," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 10, no. 7, pp. 911–921, 1991.
- [8] J. B. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.
- [9] X. W. Xu, N. Yuruk, Z. D. Feng, and T. A. J. Schweiger, "SCAN: a structural clustering algorithm for networks," in *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '07)*, pp. 824–833, ACM, August 2007.
- [10] H.-W. Liu, "Community detection by affinity propagation with various similarity measures," in *Proceedings of the 4th International Joint Conference on Computational Sciences and Optimization (CSO '11)*, pp. 182–186, April 2011.
- [11] H. Cheng, Y. Zhou, X. Huang, and J. X. Yu, "Clustering large attributed information networks: an efficient incremental computing approach," *Data Mining and Knowledge Discovery*, vol. 25, no. 3, pp. 450–477, 2012.
- [12] K. Wagstaff, C. Cardie, S. Rogers, S. Schrödl et al., "Constrained K-means clustering with background knowledge," in *Proceedings of the 18th International Conference on Machine Learning (ICML '01)*, pp. 577–584, Morgan Kaufmann Publishers, 2001.
- [13] K. Wagstaff and C. Cardie, "Clustering with instance-level constraints," in *Proceedings of the 17th International Conference on Machine Learning (ICML '00)*, pp. 1103–1110, Morgan Kaufmann Publishers, 2000.
- [14] M. Bilenko, S. Basu, and R. J. Mooney, "Integrating constraints and metric learning in semisupervised clustering," in *Proceedings of the 21st International Conference on Machine Learning (ICML '04)*, pp. 81–88, ACM, July 2004.
- [15] D. Klein, S. D. Kamvar, and C. D. Manning, "From instance-level constraints to space-level constraints: making the most of prior knowledge in data clustering," in *Proceedings of the 19th International Conference on Machine Learning (ICML '02)*, pp. 307–314, Morgan Kaufmann Publishers, 2002.
- [16] W. Tang, H. Xiong, S. Zhong, and J. Wu, "Enhancing semisupervised clustering: a feature projection perspective," in *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '07)*, pp. 707–716, ACM, August 2007.
- [17] Y. Xiao and J. Yu, "Semisupervised clustering based on affinity propagation algorithm," *Journal of Software*, vol. 19, no. 11, pp. 2803–2813, 2008.
- [18] C. Ruiz, M. Spiliopoulou, and E. Menasalvas, "C-DBSCAN: density-Based clustering with constraints," in *Proceedings of the 11th Rough Sets, Fuzzy Sets, Data Mining and Granular Computing*, pp. 216–223, Springer, 2007.
- [19] I. E. Givoni and B. J. Frey, "Semisupervised Affinity Propagation with Instance-Level Constraints," in *Proceedings of the 12th International Conference on Artificial Intelligence and Statistics*, pp. 161–168, JMLR, 2009.
- [20] A. Bar-Hillel, T. Hertz, N. Shental, and D. Weinshall, "Learning Distance Functions using Equivalence Relations," in *Proceedings of the 20th International Conference on Machine Learning*, pp. 11–18, ACM, August 2003.
- [21] S. Basu, M. Bilenko, and R. J. Mooney, "A probabilistic framework for semisupervised clustering," in *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '04)*, pp. 59–68, August 2004.
- [22] S. D. Kamvar, D. Klein, and C. Manning, "Spectral learning," in *Proceedings of the 18th International Joint Conference on Artificial Intelligence*, pp. 561–566, ACM, 2003.
- [23] X. Wang and I. Davidson, "Flexible constrained spectral clustering," in *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '10)*, pp. 563–572, ACM, July 2010.
- [24] W. F. Chen and G. C. Feng, "Spectral clustering: a semisupervised approach," *Neurocomputing*, vol. 77, no. 1, pp. 229–242, 2012.
- [25] B. Kulis, S. Basu, I. Dhillon, and R. Mooney, "Semisupervised graph clustering: a kernel approach," in *Proceedings of the 22nd International Conference on Machine Learning (ICML '05)*, pp. 457–464, ACM, August 2005.
- [26] X. K. Ma, L. Gao, X. R. Yong, and L. Fu, "Semisupervised clustering algorithm for community structure detection in complex networks," *Physica A*, vol. 389, no. 1, pp. 187–197, 2010.
- [27] G. Jeh and J. Widom, "SimRank: a measure of structural-context similarity," in *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '02)*, pp. 538–543, ACM, July 2002.
- [28] B. F. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 305, no. 5814, pp. 972–976, 2007.
- [29] A. Lancichinetti and S. Fortunato, "Limits of modularity maximization in community detection," *Physical Review E*, vol. 84, no. 6, Article ID 066122, 2011.
- [30] S. White and P. Smyth, "A spectral clustering approach to finding communities in graphs," in *Proceedings of SIAM Conference on Data Mining (SDM '05)*, pp. 76–84, 2005.
- [31] J. Q. Jiang, A. W. M. Dress, and G. Yang, "A spectral clustering-based framework for detecting community structures in complex networks," *Applied Mathematics Letters*, vol. 22, no. 9, pp. 1479–1482, 2009.
- [32] I. S. Dhillon, Y. Guan, and B. Kulis, "Kernel k-means, spectral clustering and normalized cuts," in *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '04)*, pp. 551–556, ACM, August 2004.

- [33] Z. P. Li, S. H. Zhang, R. S. Wang, X.-S. Zhang, and L. Chen, "Quantitative function for community detection," *Physical Review E*, vol. 77, no. 3, Article ID 036109, 9 pages, 2008.
- [34] I. E. Givoni and B. J. Frey, "A binary variable model for affinity propagation," *Neural Computation*, vol. 21, no. 6, pp. 1589–1600, 2009.
- [35] C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, New York, NY, USA, 2006.
- [36] <http://www-personal.umich.edu/~mejn/netdata/>.
- [37] P. Sen, G. M. Namata, M. Bilgic, L. Getoor, B. Gallagher, and T. Eliassi-Rad, "Collective classification in network data," *AI Magazine*, vol. 29, no. 3, pp. 93–106, 2008.
- [38] <http://www.cs.umd.edu/~sen/lbc-proj/LBC.html>.

## Research Article

# UCAV Path Planning by Fitness-Scaling Adaptive Chaotic Particle Swarm Optimization

Yudong Zhang,<sup>1</sup> Lenan Wu,<sup>2</sup> and Shuihua Wang<sup>1,3</sup>

<sup>1</sup> School of Computer Science and Technology, Nanjing Normal University, Nanjing, Jiangsu 210023, China

<sup>2</sup> School of Information Science and Engineering, Southeast University, Nanjing, Jiangsu 210096, China

<sup>3</sup> School of Electronic Science and Engineering, Nanjing University, Nanjing, Jiangsu 210046, China

Correspondence should be addressed to Shuihua Wang; [shuihuaw2007@gmail.com](mailto:shuihuaw2007@gmail.com)

Received 10 April 2013; Revised 15 June 2013; Accepted 28 June 2013

Academic Editor: Saeed Balochian

Copyright © 2013 Yudong Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Path planning plays an extremely important role in the design of UCAVs to accomplish the air combat task fleetly and reliably. The planned path should ensure that UCAVs reach the destination along the optimal path with minimum probability of being found and minimal consumed fuel. Traditional methods tend to find local best solutions due to the large search space. In this paper, a Fitness-scaling Adaptive Chaotic Particle Swarm Optimization (FAC-PSO) approach was proposed as a fast and robust approach for the task of path planning of UCAVs. The FAC-PSO employed the fitness-scaling method, the adaptive parameter mechanism, and the chaotic theory. Experiments show that the FAC-PSO is more robust and costs less time than elite genetic algorithm with migration, simulated annealing, and chaotic artificial bee colony. Moreover, the FAC-PSO performs well on the application of dynamic path planning when the threats cruise randomly and on the application of 3D path planning.

## 1. Introduction

Unmanned combat air vehicle (UCAV) is an experimental class of the unmanned aerial vehicle (UAV). UCAVs differ from ordinary UAVs because they are designed to deliver weapons to attack enemy targets. The elimination of the need for an onboard human crew in a UCAV that may be shot down over enemy territory has obvious advantages for personnel safety. In addition, much equipment necessary for a human pilot (such as the cockpit, flight controls, oxygen, and seat/ejection seat) can be omitted from an unmanned vehicle, resulting in a decrease in weight possibly allowing greater payloads, range, and maneuverability.

The path planning of UCAV is to generate a space path between an initial safe location and the desired dangerous destination that has an optimal or near-optimal performance under specific constraint conditions. It is always a complex research subject, so it is an imperative technology required in the design of UCAV. Series of algorithms have been proposed to solve this complicated multiconstrained optimization problem. Allaire used a genetic algorithm (GA) to realize the FPGA implementation for UAV real-time path planning [1].

Duan et al. proposed an improved particle swarm optimization to optimize the formation reconfiguration control of multiple UCAVs [2], proposed a hybrid metaheuristic ant colony optimization (ACO) and differential evolution (DE) to solve the UCAV three-dimension path-planning problem [3], and proposed a max-min adaptive ant colony algorithm for multi-UCAVs coordinated trajectory replanning [4]. Mou et al. proposed a modified ant colony algorithm as a fast and efficient approach for path planning of UCAV [5]. Zhang et al. proposed an improved artificial bee colony algorithm for UCAV path-planning problem [6]. However, these methods can easily be trapped into local minima and cannot solve the contradiction between the goal optimization and excessive information.

PSO is well known for its lower computational costs, and its most attractive feature is that it requires less computational bookkeeping and only a few lines of implementation codes. In order to improve the performance of a traditional PSO, three improvements are proposed: (I) a new power-rank fitness-scaling method, by which the scaled values are suitable for following selection; (II) adaptively varied parameters to search an expansive area at the prophase stage and

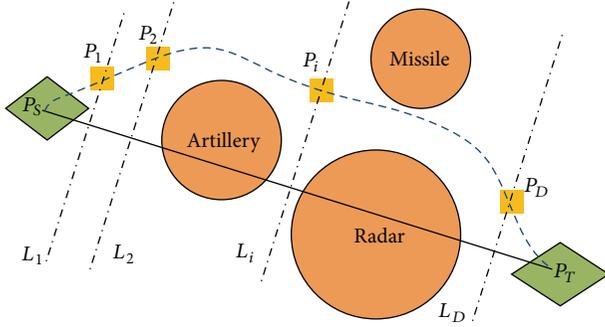


FIGURE 1: Typical 2D UCAV battlefield model.

a restricted area at the anaphase stage; and (III) introduction of chaos to improve the robustness of the basic PSO algorithm considering its outstanding performance of jumping out of stagnation.

The remainder of this paper is organized as follows. Section 2 introduces the encoding strategy for path planning. Section 3 discusses the performance evaluation function containing both the threat cost and the fuel cost. Section 4 introduces the basic principles of canonical PSO. Section 5 gives detailed description of our proposed method—Fitness-scaling Adaptive Chaotic PSO (FAC-PSO). Experiments in Section 6 compared our proposed method with elite genetic algorithm with migration, simulated annealing, and chaotic artificial bee colony. The statistical results on 100 different runs demonstrate that the FAC-PSO is superior to other algorithms with respect to success rate and computation time. Besides, we also applied our approach in the field of dynamic UCAV path planning. Final Section 7 is devoted to the conclusions.

## 2. Path Encoding

In this model, the starting point and the target point are defined as  $P_S$  and  $P_T$ , respectively. There are threatening areas in the task region, such as artillery, radar, and missile, which all are presented in the form of a circle. Inside of the threatening areas, the UCAV should be vulnerable to the threat with a certain probability proportional to the distance away from the threat center, while outside of the threatening areas, the UCAV should be safe without being attacked. The task of path planning is to design an optimal path between start point and target point considering all these threatening areas as shown in Figure 1.

We connect the starting point and target point and then divide the straight line  $P_S P_T$  into  $(D + 1)$  equal portions. At each segment point, draw the vertical line of  $P_S P_T$ , which can be labeled with  $L_1, L_2, \dots, L_i, \dots, L_D$ . Select discrete points  $P_i$  at each  $L_i$ . In this way, the path from the starting node to the target node can be described as follows:

$$\text{path} = \{P_S, P_1, P_2, \dots, P_i, \dots, P_D, P_T\}. \quad (1)$$

The location of  $P_S$  and  $P_T$  is known for UCAV, and the location of line  $L_i$  ( $i = 1, 2, \dots, D$ ) can be easily calculated. Therefore, each point  $P_i$  ( $i = 1, 2, \dots, D$ ) can be expressed

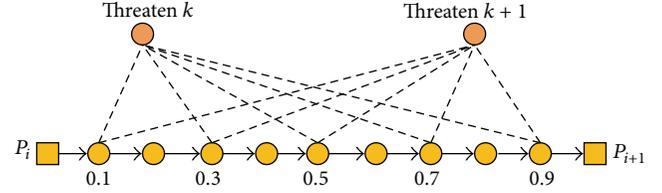


FIGURE 2: Computation of threat cost.

using only 1 parameter, namely, its distance to the straight line  $P_S P_T$ . In a word, there are total  $D$  parameters in (1), so our task is a  $D$ -dimensional optimization problem. In the following section, we let  $P_0 = P_S$  and  $P_{D+1} = P_T$  in order to simplify the expression.

## 3. Performance Function

The performance indicators of the planned path mainly consist of the threat cost  $J_T$  and the fuel cost  $J_F$ . Their calculation formulas are expressed as follows:

$$J_T = \sum_{i=0}^D J_T(i), \quad (2)$$

$$J_F = \sum_{i=0}^D J_F(i).$$

Here,  $J_T(i)$  and  $J_F(i)$  denote the threat cost and fuel cost at the  $i$ th subpath from  $P_i$  to  $P_{i+1}$ , respectively. The threat cost of subpath is calculated by an approximation based on five discrete points along the subpath as shown in Figure 2. If the  $i$ th subpath ( $P_i, P_{i+1}$ ) is within the effect range, the threat cost is given as [7]

$$J_T(i) = \frac{L_i}{5} \sum_{k=1}^{N_T} T_k \left( \frac{1}{d_{0.1,i,k}^4} + \frac{1}{d_{0.3,i,k}^4} + \frac{1}{d_{0.5,i,k}^4} + \frac{1}{d_{0.7,i,k}^4} + \frac{1}{d_{0.9,i,k}^4} \right). \quad (3)$$

Here,  $N_T$  denotes the number of threatening areas,  $L_i$  denotes the length of  $i$ th subpath,  $T_k$  denotes the degree of threatening, and  $d_{0.1,i,k}$  denotes the distance from the 1/10 point on the  $i$ th subpath to the  $k$ th threat area.

Suppose that the velocity of UCAV is constant, the fuel cost of the  $i$ th subpath  $J_F(i)$  can be considered proportional to  $L_i$ . Therefore, the total cost of the path is proportional to the total length of the path  $L$ .

The total cost for traveling along the trajectory comes from a weighted sum of the threat and fuel costs, as defined in the following formula

$$J = \omega J_T + (1 - \omega) J_F, \quad (4)$$

where  $\omega$  is a variable between 0 and 1, giving the designer certain flexibility to dispose relations between the threat exposition degree and the fuel consumption. If  $\omega$  approaches 1, a safer path is needed and less attention is paid to the fuel. Alternatively, if  $\omega$  approaches 0, a shorter path is needed even on the cost of sacrifice the safety. In this study, we determine it as 0.5 by the suggestion from [8], indicating that the threat is as important as the fuel.

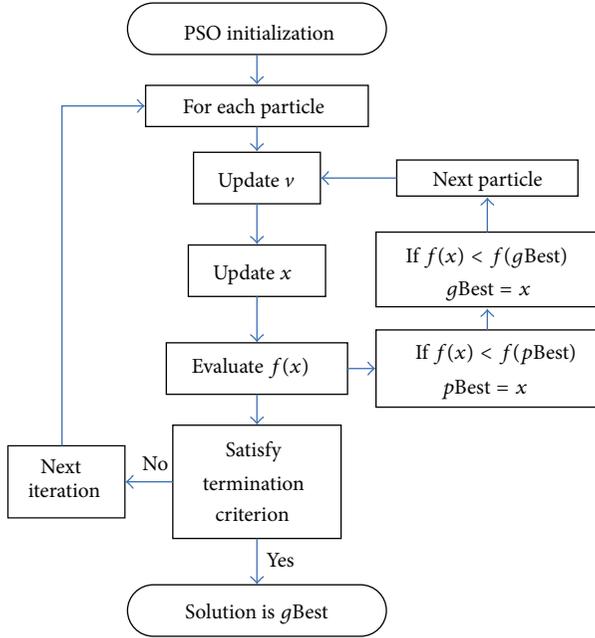


FIGURE 3: Flow chart of the PSO algorithm.

#### 4. Particle Swarm Optimization

PSO is a population-based stochastic optimization technique, which simulates the social behavior of a swarm of birds, flocking bees, and fish schooling [9]. By randomly initializing the algorithm with candidate solutions, the PSO successfully leads to a global optimum. This is achieved by an iterative procedure based on the processes of movement and intelligence in an evolutionary system. Figure 3 shows the flow chart of a PSO algorithm.

In PSO, each potential solution is represented as a particle. Two properties (position  $x$  and velocity  $v$ ) are associated with each particle. Suppose  $x$  and  $v$  of the  $i$ th particle are given as

$$\begin{aligned} x &= (x_{i1}, x_{i2}, \dots, x_{iN}), \\ v &= (v_{i1}, v_{i2}, \dots, v_{iN}), \end{aligned} \quad (5)$$

where  $N$  stands for the dimensions of the problem. In each iteration, a fitness function is evaluated for all the particles in the swarm. The velocity of each particle is updated by keeping track of the two best positions. One is the best position a particle has traversed so far and called “ $p$ Best”. The other is the best position that any neighbor of a particle has traversed so far. It is a neighborhood best called “ $n$ Best”. When a particle takes the whole population as its neighborhood, the neighborhood best becomes the global best and is accordingly called “ $g$ Best”. Hence, a particle’s velocity and position are updated as follows:

$$\begin{aligned} v &= \omega \cdot v + c_1 r_1 (p\text{Best} - x) + c_2 r_2 (n\text{Best} - x), \\ x &= x + v\Delta t, \end{aligned} \quad (6)$$

where  $\omega$  is called the “*inertia weight*” that controls the impact of the previous velocity of the particle on its current one.

The parameters  $c_1$  and  $c_2$  are positive constants called “*acceleration coefficients*”. The parameters  $r_1$  and  $r_2$  are random numbers uniformly distributed in the interval  $[0, 1]$ . These random numbers are updated every time they occur. The parameter  $\Delta t$  stands for the given time step.

The population of particles is then moved according to (6) and tends to cluster together from different directions. However, a maximum velocity  $v_{\max}$  should not be exceeded by any particle to keep the search within a meaningful solution space. The PSO algorithm runs through these processes iteratively until the termination criterion is satisfied [10].

#### 5. Principle of FAC-PSO

The PSO has proven to perform better than GA, DE, and ACO [11]. However, we can make further improvements from the following three aspects.

**5.1. Fitness Scaling.** Fitness scaling converts the raw fitness scores that are returned by the fitness function to values in a range that is suitable for the selection function [12]. The selection function uses the scaled fitness values to select the particles of the next generation. Then, the selection function assigns a higher probability of selection to particles with higher scaled values.

There exist bundles of fitness-scaling methods. One of the most common scaling techniques is traditional linear scaling, which remaps the fitness values of each particle using the following equation:

$$f_{\text{linear}} = a + b \times f_{\text{raw}}, \quad (7)$$

where  $a$  and  $b$  are constants defined by users. Another option is the rank scaling, which is obtained by sorting all the bees by their raw fitness values

$$f_{\text{rank}} = r, \quad (8)$$

where  $r$  denotes the rank of the individual particle. The third option is the power scaling method which is instead computed with

$$f_{\text{power}} = f_{\text{raw}}^k, \quad (9)$$

where  $k$  is a problem-dependent exponent that might require change during a run to stretch or shrink the range as needed. Top scaling is the 4th option and probably the simplest scaling method. Using this approach, several of the top individuals have their fitness set to the same value, with all remaining individuals having their fitness values set to zero. This simple concept yields

$$f_{\text{top}} = \begin{cases} s & f_{\text{raw}} \geq c \\ 0 & f_{\text{raw}} < c, \end{cases} \quad (10)$$

where  $s$  is the user-defined constant,  $c$  is the threshold.

Among those fitness-scaling methods, the power scaling finds a solution nearly the most quickly due to improvement of diversity but it suffers from instability [13]; meanwhile,

the rank scaling shows stability on different types of tests [14]. Therefore, a new power-rank scaling method was proposed combing both power and rank strategies as follows:

$$fit_i = \frac{r_i^k}{\sum_{i=1}^N r_i^k}, \quad (11)$$

where  $r_i$  is the rank of  $i$ th individual bee,  $N$  is the number of population. Our strategy contains a three-step process. First, all bees are sorted to obtain the corresponding ranks. Second, powers are computed for exponential values  $k$ . Third, the scaled values are normalized by dividing the sum of the scaled values over the entire population.

**5.2. Adaptive Parameters.** Another improvement lies in changing the parameters ( $\omega, c_1, c_2$ ) adaptively. In the search process of PSO, the search space will gradually reduce as the generation increases. Therefore, we hope to search an expansive area with low precision at the prophase stage while searching a restricted area with high precision at the anaphase stage as listed in Table 1. The detailed formulas of those adaptive parameters are as follows:

$$\begin{aligned} \omega &= \omega_i - \frac{\omega_i - \omega_f}{\text{MaxGeneration}} * \text{Generation} \quad (\omega_i > \omega_f), \\ c_1 &= c_{1i} - \frac{c_{1i} - c_{1f}}{\text{MaxGeneration}} * \text{Generation} \quad (c_{1i} > c_{1f}), \\ c_2 &= c_{2i} - \frac{c_{2i} - c_{2f}}{\text{MaxGeneration}} * \text{Generation} \quad (c_{2i} < c_{2f}). \end{aligned} \quad (12)$$

Here, the indexes  $i$  and  $f$  denote “initial” and “final”, respectively.

**5.3. Chaotic Random Number.** The parameters ( $r_1, r_2$ ) were generated by a pseudorandom number generator (RNG) in classical PSO. The RNG cannot ensure the optimization's ergodicity in solution space because it is absolutely random; therefore, a chaotic operator was employed to generate parameters ( $r_1, r_2$ ) by the following formula:

$$r_i(t+1) = 4.0 * r_i(t) * [1 - r_i(t)], \quad i = 1, 2, \quad (13)$$

where  $r_0 \in (0, 1)$  and  $r_0 \notin \{0.25, 0.5, 0.75\}$ . A very small difference in the initial value of  $x$  would give rise to a large difference in its long-time behavior as shown in Figure 4. The track of chaotic variable  $x_n$  can travel ergodically over the whole space of interest.

Figure 5 shows that the series is in the cycle (0.75) when initial points are 0.25 or 0.75 since  $4 * 0.25 * (1 - 0.25) = 0.75$  and  $4 * 0.75 * (1 - 0.75) = 0.75$ . The series is in cycle (0) when initial point is 0.5 since  $4 * 0.5 * (1 - 0.5) = 1$  and  $4 * 1 * (1 - 1) = 0$ . Therefore, those three points (0.25, 0.5, and 0.75) will make the series lose chaotic property.

## 6. Experiments

The experiments were carried out on the platform of P4 IBM with 3 GHz main frequency and 2 GB memory, running

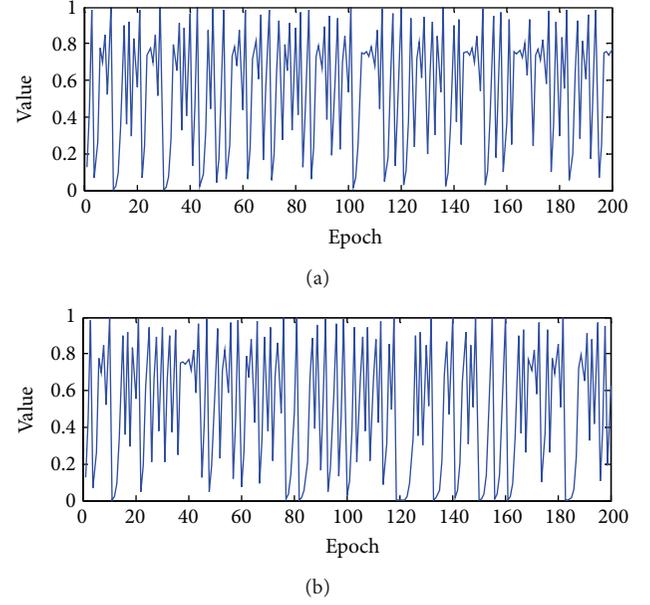


FIGURE 4: Chaotic Property of logistic equation: (a)  $x_0 = 0.12345678$ ; (b)  $x_0 = 0.12345679$ .

under Windows XP operating system. The algorithm was developed via the global optimization toolbox of MatLab 2011a.

**6.1. Threat Setting.** Set the coordinates of the starting point as (5, 5) and the target point as (100, 100). In the flight course, there exist eight threat areas listed in Table 2. Suppose that the codes of initial path are all zeros, which corresponds to a straight line from starting point directly to the target point as shown in Figure 6. Traditional gradient-based methods will guide the 12th–20th nodes of the path to search the upper-left area, and they will finally be misled into the local minima. However, the evolutionary algorithms including GA, PSO, and our proposed FAC-PSO are able to jump from the local minima and search the bottom-right area, where the global minimal point locates in.

**6.2. Algorithm Comparison.** We compared the proposed FAC-PSO method with elite genetic algorithms with migration (EGAM) [15], simulated annealing (SA) [16], chaotic artificial bee colony (CABC) [8], and standard PSO. The parameters are obtained through trial-and-error method and shown in Table 3. Here,  $NP$  means the number of populations/bees/particles corresponding to different algorithms, and  $MaxEpoch$  means the maximum iterative epochs.  $P_c, P_m,$  and  $P_e$  of EGAM stand for the crossover probability, mutation probability, and elite probability, respectively.  $TDF, T_I,$  and  $T_F$  of SA denote the temperature decrease function, initial temperature, and final temperature, respectively.  $N_F$  in CABC represents the number of foods.

Each algorithm ran 100 times, and the success rate was calculated and shown in Table 4. It indicates that the proposed FAC-PSO show slight superiority to other algorithms when

TABLE 1: Parameters variation.

	$\Omega$	$c_1$	$c_2$	Performance
Prophase	Larger	Larger	Smaller	PSO searches for global optima in an expansive area with low precision
Anaphase	Smaller	Smaller	Larger	PSO searches for local optima in a limited area with high precision

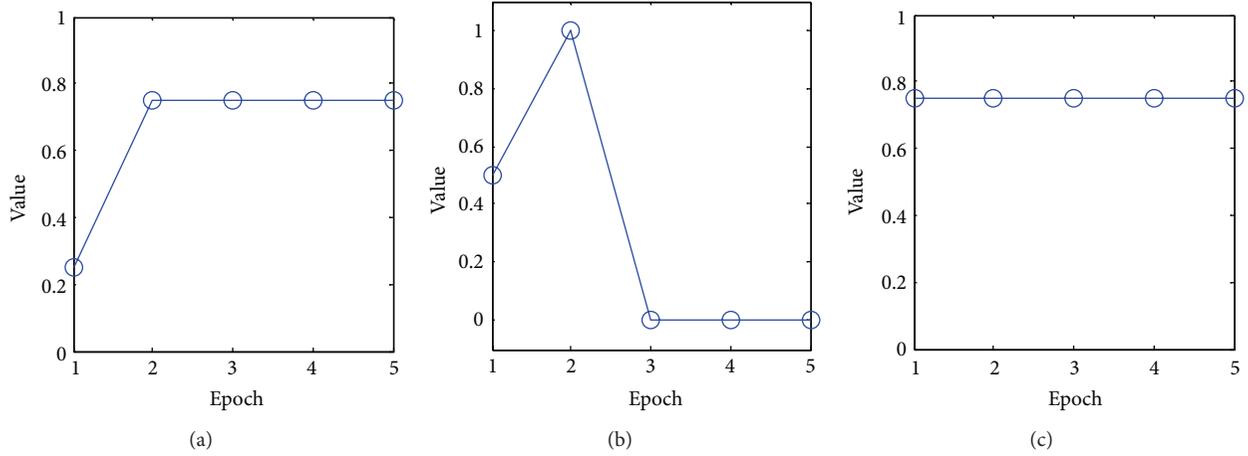


FIGURE 5: Loss of chaotic property at specific initial values: (a)  $x_0 = 0.25$ ; (b)  $x_0 = 0.5$ ; and (c)  $x_0 = 0.75$ .

TABLE 2: Information of 2D threatening objects.

Index	Position	Radius
1	(10, 30)	14
2	(10, 50)	10
3	(20, 80)	20
4	(40, 15)	12
5	(40, 50)	15
6	(50, 70)	12
7	(75, 70)	14
8	(80, 40)	12

$D = 10$ . As the  $D$  increases, the FAC-PSO shows more robustness compared to other algorithms. It should be noted that a larger  $D$  makes the search space larger, which leads to the success rate of all the algorithms decreasing.

We take  $D = 20$  as an example, choose a typical run and show the convergence plot in Figure 7. It indicates that the proposed FAC-PSO was trapped into local minima at about 42 epochs but it jump out at about 45th epoch. Conversely, the EGAM, SA, CABG, and PSO were stagnated in the local minima over all 100 epochs.

The final searched paths of the four algorithms are shown in Figure 8. It indicates that our proposed FAC-PSO found the global best path of the algorithm, while the other four algorithms failed at this run.

6.3. Time Comparison. Computation time is another important factor used to evaluate the algorithm. Since a failed run usually takes little time due to early stagnation, we only

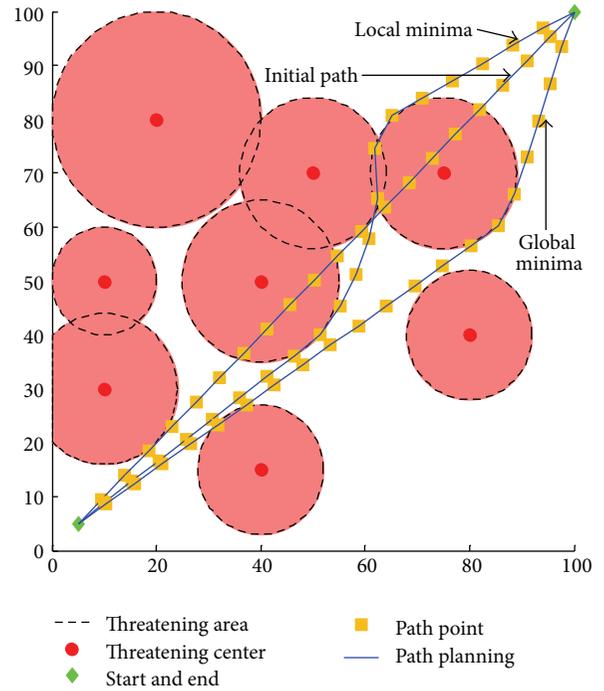


FIGURE 6: Local minima and global minima.

consider the success runs. The average computation times of each algorithm of different sizes of  $D$  are shown in Table 5. It indicates that the proposed algorithm takes the least time in spite of the size of  $D$ ; moreover, the SA takes the second least time for calculation. The most time-consuming algorithm is the CABG.

TABLE 3: Parameters of algorithms.

Algorithm	Parameter setting
EGAM	$NP = 20$ ; $MaxEpoch = 100$ ; $P_C = 0.8$ ; $P_m = 0.1$ ; $P_e = 0.1$ ; and $migration\ interval = 20$ .
SA	$NP = 20$ ; $MaxEpoch = 100$ ; TDF = "exponential"; $T_I = 100$ , and $T_F = 0$ .
CABC	$NP = 20$ ; $MaxEpoch = 100$ ; and $N_F = 10$ .
PSO	$NP = 20$ ; $MaxEpoch = 100$ ; $v_{max} = 1$ ; $\omega = 0.6$ ; $c_1 = 1$ ; and $c_2 = 1$ .
FAC-PSO	$NP = 20$ ; $MaxEpoch = 100$ ; $v_{max} = 1$ ; $\omega_i = 0.9$ ; $\omega_f = 0.4$ ; $c_{1i} = 2.5$ ; $c_{1f} = 0.5$ , $c_{2i} = 0.5$ ; and $c_{2f} = 2.5$ .

TABLE 4: Success rates of different algorithms for 2D UCAV.

$D$	EGAM	SA	CABC	PSO	FAC-PSO
10	78%	22%	80%	75%	<b>87%</b>
15	66%	12%	67%	71%	<b>85%</b>
20	53%	3%	59%	56%	<b>80%</b>

TABLE 5: Average computation time (s).

$D$	EGAM	SA	CABC	PSO	FAC-PSO
10	12.3	11.6	14.6	13.0	<b>10.2</b>
15	12.9	13.8	15.3	14.7	<b>11.3</b>
20	14.5	13.6	16.8	14.9	<b>13.7</b>

TABLE 6: Information of 3D threatening objects.

Index	Position	Radius
1	(10, 30, 30)	14
2	(10, 50, 20)	10
3	(20, 80, 40)	20
4	(40, 15, 70)	12
5	(40, 50, 50)	15
6	(50, 70, 40)	12
7	(75, 70, 35)	14
8	(80, 40, 50)	12
9	(40, 55, 30)	10
10	(30, 40, 40)	15

**6.4. Dynamic Path Planning.** The aforementioned paths are static ones which are determined by a beforehand-known map and threat information, but the UCAV usually meets unforeseen threats in actual flight course, so they must possess dynamic path-planning ability. When UCAV detects instantaneous or moving threat, it must replan the path so as to avoid the new arisen threat by revising the former path.

Suppose that all the threatening areas can move randomly, then, the flight path of UCAV should change after each move according to the current threat positions. The UCAV paths by the proposed FAC-PSO at steps 0, 5, 10, and 15 are shown in Figure 9, which imply the feasibility of FAC-PSO under moving threatening conditions.

**6.5. 3D Path Planning.** We applied our method to 3D UCAV path planning. First, we generate a cube ( $100 \times 100 \times 100$ ). Second, we generated 10 threats, and their coordinates and radii are listed in Table 6.

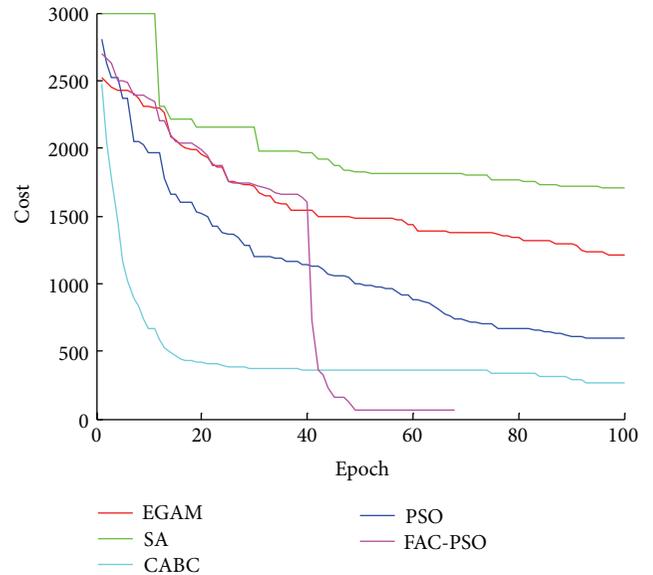


FIGURE 7: A typical convergence plot of different algorithms.

TABLE 7: Success rates of different algorithms for 3D UCAV.

$D$	EGAM	SA	CABC	PSO	FAC-PSO
15	65%	9%	64%	65%	<b>74%</b>
20	41%	2%	43%	42%	<b>68%</b>
25	16%	0%	21%	20%	<b>53%</b>

We set the coordinates of the starting point as (5, 5, and 5) and the target point as (100, 100, and 100). We compared the proposed FAC-PSO method with EGAM, SA, CABC, and PSO. All parameters are the same as Table 3 except that the maximal epoch is changed to 1000. Each algorithm ran 100 times, and the success rate was calculated and shown in Table 7. We found that the FAC-PSO performs best among all algorithms for 3D UCAV path planning.

## 7. Conclusions

In this study, a novel FAC-PSO approach for UCAV path planning was proposed. We first investigate the path encoding strategy and then construct the cost function which combines the threat cost and fuel cost simultaneously. The FAC-PSO algorithm was proposed utilizing the fitness-scaling, the adaptive mechanism, and the ergodicity and irregularity of the chaos. Compared to standard PSO, the FAC-PSO is more

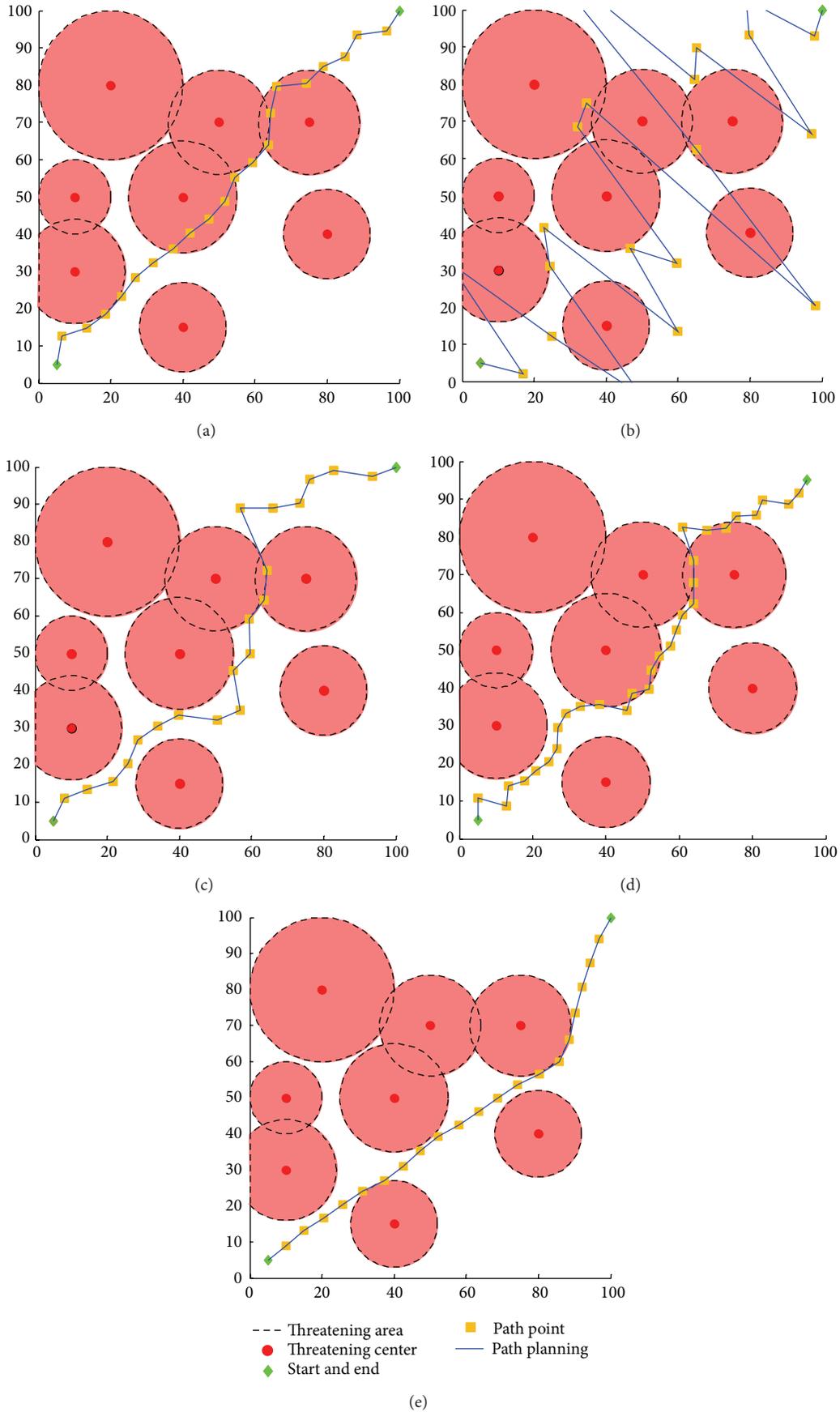


FIGURE 8: Planned path of (a) EGAM; (b) SA; (c) CABC; (d) PSO; and (e) FAC-PSO.

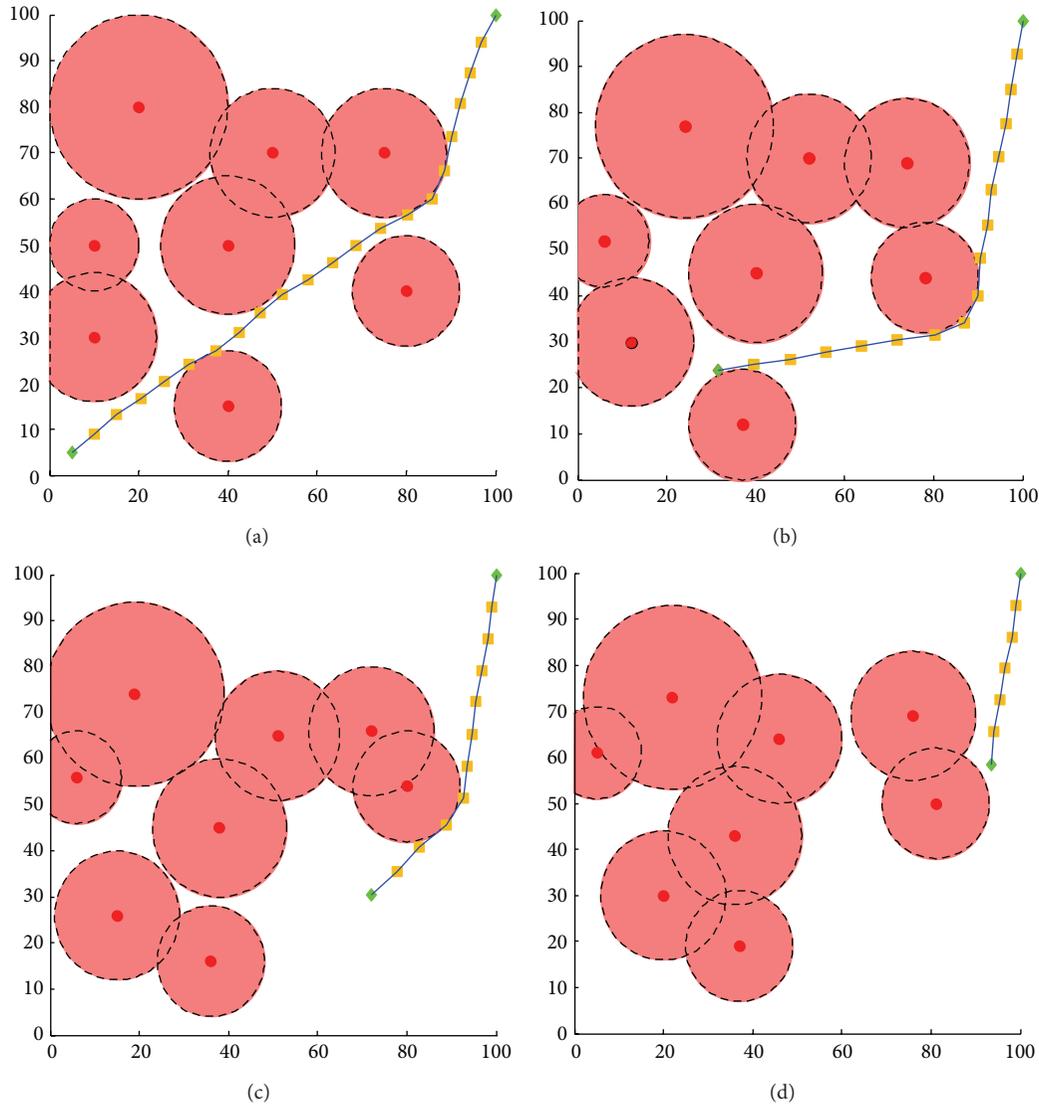


FIGURE 9: All threatening obstacles are moving for dynamic path planning: (a) step 0; (b) step 5; (c) step 10; and (d) step 15.

powerful at jumping out of local minima as well as speeding up the procedures of finding the global optimal minima.

The simulation results show that the proposed FAC-PSO excels EGAM, SA, CABC, and PSO algorithms with respect to success rate and computation time. We extended our experiment to 2D dynamic path planning and 3D path planning. All prove the superiority of FAC-PSO. Therefore, it is a feasible and effective way for UCAV path planning.

## References

- [1] F. C. J. Allaire, M. Tarbouchi, G. Labonté, and G. Fusina, "FPGA implementation of genetic algorithm for UAV real-time path planning," *Journal of Intelligent and Robotic Systems*, vol. 54, no. 1–3, pp. 495–510, 2009.
- [2] H.-B. Duan, G.-J. Ma, and D.-L. Luo, "Optimal formation reconfiguration control of multiple UCAVs using improved particle swarm optimization," *Journal of Bionic Engineering*, vol. 5, no. 4, pp. 340–347, 2008.
- [3] H. Duan, Y. Yu, X. Zhang, and S. Shao, "Three-dimension path planning for UCAV using hybrid meta-heuristic ACO-DE algorithm," *Simulation Modelling Practice and Theory*, vol. 18, no. 8, pp. 1104–1115, 2010.
- [4] H.-B. Duan, X.-Y. Zhang, J. Wu, and G.-J. Ma, "Max-Min adaptive ant colony optimization approach to multi-UAVs coordinated trajectory replanning in dynamic and uncertain environments," *Journal of Bionic Engineering*, vol. 6, no. 2, pp. 161–173, 2009.
- [5] C. Mou, W. Qing-xian, and J. Chang-sheng, "A modified ant optimization algorithm for path planning of UCAV," *Applied Soft Computing Journal*, vol. 8, no. 4, pp. 1712–1718, 2008.
- [6] Y. Zhang, L. Wu, and S. Wang, "UCAV path planning based on FSCABC," *Information*, vol. 14, no. 3, pp. 687–692, 2011.
- [7] D. Rodic and A. P. Engelbrecht, "Social networks in simulated multi-robot environment," *International Journal of Intelligent Computing and Cybernetics*, vol. 1, no. 1, pp. 110–127, 2008.
- [8] C. Xu, H. Duan, and F. Liu, "Chaotic artificial bee colony approach to Uninhabited Combat Air Vehicle (UCAV) path

- planning,” *Aerospace Science and Technology*, vol. 14, no. 8, pp. 535–541, 2010.
- [9] Y. Zhang, Y. Jun, G. Wei, and L. Wu, “Find multi-objective paths in stochastic networks via chaotic immune PSO,” *Expert Systems with Applications*, vol. 37, no. 3, pp. 1911–1919, 2010.
- [10] Y. Zhang, L. Wu, G. Wei, J. Yan, and Q. Zhu, “Chaotic immune particle swarm optimization for multi-exponential fitting,” *Journal of Southeast University (Natural Science Edition)*, vol. 39, no. 4, pp. 678–683, 2009.
- [11] O. Begambre and J. E. Laier, “A hybrid Particle Swarm Optimization—simplex algorithm (PSOS) for structural damage identification,” *Advances in Engineering Software*, vol. 40, no. 9, pp. 883–891, 2009.
- [12] E. R. Tkaczyk, K. Mairing, A. H. Tkaczyk et al., “Control of the blue fluorescent protein with advanced evolutionary pulse shaping,” *Biochemical and Biophysical Research Communications*, vol. 376, no. 4, pp. 733–737, 2008.
- [13] A. M. Korsunsky and A. Constantinescu, “Work of indentation approach to the analysis of hardness and modulus of thin coatings,” *Materials Science and Engineering A*, vol. 423, no. 1–2, pp. 28–35, 2006.
- [14] Y. Wang, B. Li, and T. Weise, “Estimation of distribution and differential evolution cooperation for large scale economic load dispatch optimization of power systems,” *Information Sciences*, vol. 180, no. 12, pp. 2405–2420, 2010.
- [15] T. Kellegöz, B. Toklu, and J. Wilson, “Elite guided steady-state genetic algorithm for minimizing total tardiness in flowshops,” *Computers and Industrial Engineering*, vol. 58, no. 2, pp. 300–306, 2010.
- [16] B. Agard and B. Penz, “A simulated annealing method based on a clustering approach to determine bills of materials for a large product family,” *International Journal of Production Economics*, vol. 117, no. 2, pp. 389–401, 2009.

## Research Article

# CFSO<sup>3</sup>: A New Supervised Swarm-Based Optimization Algorithm

**Antonino Laudani, Francesco Riganti Fulginei, Alessandro Salvini,  
Maurizio Schmid, and Silvia Conforto**

*Department of Engineering, Roma Tre University, Via V. Volterra 62, 00146 Rome, Italy*

Correspondence should be addressed to Antonino Laudani; [alaudani@uniroma3.it](mailto:alaudani@uniroma3.it)

Received 3 May 2013; Accepted 7 July 2013

Academic Editor: Orwa Jaber Housheya

Copyright © 2013 Antonino Laudani et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We present CFSO<sup>3</sup>, an optimization heuristic within the class of the swarm intelligence, based on a synergy among three different features of the Continuous Flock-of-Starlings Optimization. One of the main novelties is that this optimizer is no more a classical numerical algorithm since it now can be seen as a continuous dynamic system, which can be treated by using all the mathematical instruments available for managing state equations. In addition, CFSO<sup>3</sup> allows passing from stochastic approaches to supervised deterministic ones since the random updating of parameters, a typical feature for numerical swarm-based optimization algorithms, is now fully substituted by a supervised strategy: in CFSO<sup>3</sup> the tuning of parameters is *a priori* designed for obtaining both exploration and exploitation. Indeed the exploration, that is, the escaping from a local minimum, as well as the convergence and the refinement to a solution can be designed simply by managing the eigenvalues of the CFSO state equations. Virtually in CFSO<sup>3</sup>, just the initial values of positions and velocities of the swarm members have to be randomly assigned. Both standard and parallel versions of CFSO<sup>3</sup> together with validations on classical benchmarks are presented.

## 1. Introduction

Exploration and exploitation are the two fundamental requirements for the algorithms devoted to inverse problems and/or optimization. In many applications more than one global optimum could exist, and/or the space of the solutions to be investigated could show a very large dimension. In all these cases, the risk of remaining entrapped into a local minimum is quite high, especially if the algorithm in use is not able to explore large spaces [1]. On the other hand, in other cases, it is necessary to converge to a solution granting a high level of accuracy. These conditions are difficult to be satisfied if the algorithm is not able to perform good exploitation [2]. Thus, apparently, exploration and exploitation seem to be antithetic requirements. With the aim to match these two properties in only one strategy of optimization, we aim to present in this paper CFSO<sup>3</sup> that takes its main inspiration from swarm algorithms and in particular from their translation into continuous dynamic systems.

From a historical point of view, the swarm optimization has been introduced by Kennedy and Eberhart [3] in the 1990s with the Particle Swarm Optimization (PSO),

a heuristic inspired to the social and collective behaviour shown by several animal species such as flock of birds or school of fishes [4, 5]. Virtually, it consists of a one-to-one correspondence between the motion of flocks searching for food and the iterative steps of algorithms searching for the best solution for optimization. Many authors have published a large quantity of works related to PSO, and the vastness of applications into different fields of science, such as engineering, physics, chemistry, artificial intelligence, and economics, testifies its success among many scientific communities (e.g., see [6–9] and the references within). A large series of changes from the original PSO have been proposed in order to improve its performances. In particular, many works focused on the way to manage the tuning of parameters for achieving better convergence to the global optimum and/or for improving exploration for multimodal problems (e.g., see [10–19] and the reference within), and the effects of topological rules on performances have been discussed [20–26]. The basic idea of topological rules is to link each member of the swarm with others by generating more complex information exchange among particles than the simple use of the global best. A particular way to link the particle behaviors is achieved

by using a kind of neighbor-velocity matching, that is, the idea of the Flock-of-Starlings Optimization (FSO) [27, 28], inspired by a naturalistic work presented in [29]. The FSO uses topology for exchanging the information about the current velocity of each connected particle/bird. Recently, many authors have proposed several attempts to treat the swarm numerical algorithms as continuous dynamical systems. For example, important contributions are in [30–32]. The main goal of these works is to investigate the stability of PSO and the effects on the optimization performance produced by different settings of the parameters. In this scenario, the authors of the present paper have translated the numerical FSO equations into the state equations of a kinetic dynamic system in the time domain (continuum), by proposing the so-called Continuous Flock-of-Starlings Optimization (CFSO) [33–35]. This new optimizer, that is, the starting point of the present paper, turns out to be quite effective, thanks to the use of mathematical closed forms which describe the swarm member trajectories as function of time, since it opens the road to all those mathematical instruments used for studying state equations. In addition, CFSO has several features that make it very interesting and promising such as, among other ones, the setting of parameters without using randomness. Thanks to the stability analysis of the CFSO state equations, the tuning of the parameters becomes the way for controlling the stability of the trajectories within the space of solutions, that is, for controlling the exploration and the exploitation, since the parameter values fix the values of the eigenvalues of the dynamic system (e.g., see [35–38] and the reference within).

In this paper, results derived from investigation on this distinguishing behaviour of the CFSO are discussed, in order to verify the effectiveness of the innovative nonstochastic approach characterizing this continuous swarm optimizer. In particular, the present paper shows a hybridization of three different features of CFSO, which allow us to enhance both exploration and convergence. Indeed, the changing from exalting exploration to giving a boost to the convergence/refinement is achievable simply by means of a suitable setting of the real part of pole values (or the eigenvalues) related to the CFSO state equations: in particular *pure imaginary poles* perform the exploration of the whole space of solutions; *instable poles* allow to escape from local minima, whereas refinement of the solution is obtained by launching an *asymptotically stable* CFSO. Only the initial position and velocity of each individual (birds) are randomly assigned.

Validations on classical benchmarks in order to show the advantages of the proposed continuous approach are presented, and the CFSO implementation has been made available for downloading at the link indicated in [39].

## 2. Recall of the Continuous Flock-of-Starlings Optimization: From Numerical Algorithms to Dynamic System State Equations

In [33, 35] it has been proved that it is possible to convert the numerical swarm optimization algorithm, PSO or FSO, into continuous time-domain dynamical kinetic systems

described by a set of state equations. Virtually, the state equations equivalent to the rules used by the swarm algorithms for updating velocity and position of generic  $k$ th particle are

$$\begin{aligned} \dot{v}_k^j(t) &= \omega v_k^j(t) + \lambda (p_{\text{best}_k}^j(t) - x_k^j(t)) \\ &+ \gamma (g_{\text{best}}^j(t) - x_k^j(t)) + \sum_{m=1}^N h_{km} v_m^j(t), \end{aligned} \quad (1)$$

$$\dot{x}_k^j(t) = v_k^j(t), \quad (2)$$

where  $j = 1 \cdots \Delta$ ,  $\Delta$  is the dimension of the solution space, and  $\omega$ ,  $\lambda$ , and  $\gamma$  are the so-called inertial, cognitive, and social coefficients, respectively;  $g_{\text{best}}^j(t)$  is the  $j$ th component of the global best of the whole swarm, whereas  $p_{\text{best}_k}^j(t)$  is the  $j$ th component of the personal best of the  $k$ th particle;  $\sum_{m=1}^N h_{km} v_m^j(t)$  ( $h_{km} = h$  if the  $k$ th bird is controlling the  $m$ th one,  $h_{km} = 0$  otherwise) is the term that transforms the PSO into the FSO. This term, which strongly modifies the collective behaviour of the flock as shown in [27, 28], comes from the observation [29] that, in real flock each generic bird controls and follows the flight of a number  $N_{\text{ctrl.birds}}$  of other members of the flock, no matter what are their positions inside the flock. Finally, the personal best and global best have been assumed to be the excitations of the dynamic system by writing the quantity

$$\mathfrak{F}_k^j(t) = \lambda \cdot p_{\text{best}_k}^j(t) + \gamma \cdot g_{\text{best}}^j(t). \quad (3)$$

By posing  $\mu = \lambda + \gamma$  it is possible to rewrite (1) in the following form:

$$\dot{v}_k^j(t) = \omega v_k^j(t) - \mu x_k^j(t) + \sum_{m=1}^N h_{km} v_m^j(t) + \mathfrak{F}_k^j(t). \quad (4)$$

Equations (2) and (4) describe the state equations of a continuous dynamic system. Thus, if we consider a flock made of  $N$  birds, the state-equation system of the CFSO has dimension  $2N \times 2N$ , and it can be expressed, for each component related to the  $j$ th dimension, as follows (from now on, in order to simplify the notation the apex  $j$  will be dropped; consequently, all expressions must be implicitly assumed to be valid for each  $j$ th generic component):

$$\begin{bmatrix} \dot{\mathbf{v}} \\ \dot{\mathbf{x}} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{1,1} & \mathbf{A}_{1,2} \\ \mathbf{I} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{v} \\ \mathbf{x} \end{bmatrix} + \begin{bmatrix} \mathbf{F} \\ \mathbf{0} \end{bmatrix}. \quad (5)$$

Submatrix  $\mathbf{I}$  appearing in (5) is the  $N \times N$  identity matrix.  $N \times N$  square submatrix  $\mathbf{A}_{1,1}$  is defined as

$$\mathbf{A}_{1,1} = \omega \cdot \mathbf{1} + \mathbf{H}. \quad (6)$$

Matrix  $\mathbf{H}$  takes into account the neighbor-velocity-matching rule and for this reason is called neighbor-velocity-matching matrix. The nonzero values of  $\mathbf{H}$  are the nondiagonal entries for which  $h_{km} = h$  this means that the  $k$ th bird is controlling the velocity of the  $m$ th one. It is evident that, just considering the absence of the matrix  $\mathbf{H}$  in (6),

it is possible to commutate the CFSO to a continuous PSO (CPSO). Finally, the last  $N \times N$  submatrix appearing in (5) is

$$\mathbf{A}_{1,2} = -\mu \cdot \mathbf{1}, \quad (7)$$

whereas vector  $\mathbf{F}$  has each  $k$ th row-entry  $\mathfrak{F}_k(t)$  of (3).

*2.1. Solution of the CFSO State Equations.* Let us assume that the space where the solution lies is bounded, and then, also the global and personal best positions will be bounded and consequently we can apply the Laplace Transform,  $\mathcal{L}\{\bullet\}$ , in the variable  $s \in \mathbb{C}$  to the system (5) as follows:

$$s \begin{bmatrix} \mathbf{V}(s) \\ \mathbf{X}(s) \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{1,1} & \mathbf{A}_{1,2} \\ \mathbf{1} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{V}(s) \\ \mathbf{X}(s) \end{bmatrix} + \begin{bmatrix} \mathbf{F}(s) \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} \mathbf{v}(0) \\ \mathbf{x}(0) \end{bmatrix}. \quad (8)$$

In (8), both  $\mathbf{v}(0)$  and  $\mathbf{x}(0)$  are column vectors (from now on, for brevity, we will indicate  $\mathbf{x} = \mathbf{x}(0)$  and  $\mathbf{v} = \mathbf{v}(0)$ ). They take into account the initial conditions related to velocities and positions, respectively, while each single  $\mathcal{L}\{\bullet\}$  is

$$\mathbf{V}(s) = \mathcal{L}\{\mathbf{v}(t)\} = \int_0^{+\infty} \mathbf{v}(t) e^{-st} dt, \quad (9)$$

$$\mathbf{X}(s) = \mathcal{L}\{\mathbf{x}(t)\} = \int_0^{+\infty} \mathbf{x}(t) e^{-st} dt, \quad (10)$$

$$\mathbf{F}(s) = \mathcal{L}\{\mathbf{F}(t)\} = \int_0^{+\infty} \mathbf{F}(t) e^{-st} dt. \quad (11)$$

Finally, by solving (8) in the Laplace domain for  $\mathbf{X}(s)$ , we have

$$\mathbf{X}(s) = (s^2 \mathbf{1} - \mathbf{A}_{1,1} s - \mathbf{A}_{1,2})^{-1} \cdot \{(s \mathbf{1} - \mathbf{A}_{1,1}) \cdot \mathbf{x} + \mathbf{v} + \mathbf{F}\}. \quad (12)$$

Now, let us now decompose the response (solution) for the  $k$ th bird as the superposition of the free response,  $\mathbf{X}^{\text{free}}(s)$ , and the forced response,  $\mathbf{X}^{\text{forced}}(s)$ ; that is,  $\mathbf{X}(s) = \mathbf{X}^{\text{free}}(s) + \mathbf{X}^{\text{forced}}(s)$ . Thus, we have

$$\mathbf{X}^{\text{free}}(s) = (s^2 \mathbf{1} - \mathbf{A}_{1,1} s - \mathbf{A}_{1,2})^{-1} \cdot \{(s \mathbf{1} - \mathbf{A}_{1,1}) \cdot \mathbf{x} + \mathbf{v}\}, \quad (13)$$

$$\mathbf{X}^{\text{forced}}(s) = (s^2 \mathbf{1} - \mathbf{A}_{1,1} s - \mathbf{A}_{1,2})^{-1} \cdot \mathbf{F}(s). \quad (14)$$

In this way, we can collect together global and all personal bests inside the forced solution (14), whereas the initial conditions appear just in the free solution (13). This is a crucial fact since it means that, for studying the collective behaviour of the swarm, we have to analyze just the free response (13) that is not influenced by the habitat (i.e., global and personal bests valued by the fitness function), but just by the initial conditions related to each swarm member. Clearly, after having written the solution of the state equations in the Laplace domain, we have to translate the solution in the time domain. But the Laplace anti-transformation is not always possible, since the inverse of matrix  $(s^2 \mathbf{1} - \mathbf{A}_{1,1} s - \mathbf{A}_{1,2})$  could not be simply evaluated in an analytical way. Indeed,

if we write the neighbour velocity-matching, that is, matrix ( $\mathbf{H}$ ), without a pre-fixed order, the only way to solve the CFSO equations in the time domain is to adopt a numeric ordinary differential equation solver, for example, by means of a numerical Runge-Kutta algorithm (ODE suite of MATLAB). On the other hand, it has been proved in [35] that a particular ordered choice of the matrix describing the neighbour-velocity matching allows to analytically integrate the CFSO equations, by means of the inverse Laplace transforming, and closed-form expressions, in the time domain, are easily available. The fundamental advantage using closed forms is that they remarkably reduce the computational cost making easier the setup of the optimization problem. In addition they can be directly used to update position and velocity, as it will be shown in next sections.

Let us show herein the way in which it is possible to achieve closed forms. For simplicity, but without loss of generality, let us consider the so-called *fully connected* CFSO, which occurs if  $N_{\text{ctrl.birds}} = N - 1$ ; that is, each  $k$ th bird controls all the other  $N - 1$  members of the flock. This means that all the non-diagonal entries of the neighbour-velocity-matching matrix  $\mathbf{H}$  are set to a value equal to  $h$ :

$$h_{k,m} = \begin{cases} h & k \neq m \\ 0 & k = m. \end{cases} \quad (15)$$

Thus, the matrices appearing in the solution of the system (12) become, respectively,

$$s^2 \mathbf{1} - \mathbf{A}_{1,1} s - \mathbf{A}_{1,2} = \begin{pmatrix} s^2 - \omega s + \mu & -hs & \cdots & -hs \\ -hs & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & -hs \\ -hs & \cdots & -hs & s^2 - \omega s + \mu \end{pmatrix}, \quad (16)$$

$$s \mathbf{1} - \mathbf{A}_{1,1} = \begin{pmatrix} s - \omega & -h & \cdots & -h \\ -h & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & -h \\ -h & \cdots & -h & s - \omega \end{pmatrix}. \quad (17)$$

The values of the entries of the inverse,  $[C]$ , related to the symmetric matrix (16) are

$$c_{k,m} = \begin{cases} \frac{s^2 - \omega s + \mu - (N-2)hs}{(s^2 - (\omega + (N-1)h)s + \mu)(s^2 + (h-\omega)s + \mu)} & \text{if } k = m \\ \frac{hs}{(s^2 - (\omega + (N-1)h)s + \mu)(s^2 + (h-\omega)s + \mu)} & \text{if } k \neq m. \end{cases} \quad (18)$$

Thus, by compacting in one single term

$$a_k = -(\omega - h)x_k - h \sum_{m=1}^N x_m + v_k, \quad (19)$$

the solution can be finally expressed by

$$\mathbf{X}(s) = [C] \cdot [\mathbf{s}\mathbf{x} + \mathbf{a} + \mathbf{F}(s)]. \quad (20)$$

**2.2. Time Windowing.** Now, in order to solve the system of state equations (20), a further consideration must be done since the force term (vector  $\mathbf{F}$ ) is not *a priori* known: its elements depend on the trajectories followed by the birds. Consequently they must be evaluated dynamically according to the several fitness values which are monitored during the motion of the swarm/flock. Moreover, it is worth noticing that we are interested in the building of an optimizer, and, consequently, it is our aim to evaluate exactly the fitness function in correspondence with specified positions reached by each member of the flock. In order to address this goal, we subdivide the time axis into a sequence of time windows, each one having duration equal to a value  $\tau$ . In this way, we can assume, for each  $i$ th time window TW, that the excitation (3) is a constant value equal to the one evaluated at the end of the previous  $(i - 1)$ th TW. It follows that, for a generic TW $_i$  that begins at the generic time  $t_{in}$  and ends at the time  $t_{in} + \tau$ , we assume that the  $k$ th entries of the force terms (3) are equal to a constant value  $\mathfrak{F}_{k,i}(t_{in})$ , that is, for any time value within that  $i$ th window. It is worth noticing that when  $\tau \rightarrow 0$  the CFSO system would be really continuum, but the fitness function should be evaluated infinite times. Anyway, for finite values of  $\tau$ , the system (5) is still time invariant within each time window. As a consequence, it is still possible to use Laplace transforms in the previously described way. Now, the only difference is that we will obtain analytical closed forms, for the position and the velocity of each bird of the flock, valid just for a TW. The final solution on the desired time of observation will be obtained by the union of the solutions returned TW-by-TW, with the continuity being guaranteed by the initial conditions that are always evaluated at the beginning of each new TW.

**2.3. Closed Forms for the Continuous-Flock-of-Starlings Optimization.** On the basis of the previous assumptions, the Laplace transform related to (3), valid for the  $k$ th particle/bird, is (and clearly for the generic component  $j$ th) within the  $i$ th TW:

$$F_{k,i}(s) = \frac{\gamma g_{best_i} + \lambda p_{best_{k,i}}}{s}. \quad (21)$$

By elaborating and reordering (20), we obtain the following explicit expression for the Laplace transform of each  $k$ th member of the flock:

$$\begin{aligned} X_k(s) &= \frac{s x_k + a_k}{s^2 + (h - \omega) s + \mu} \\ &+ h \frac{s^2 \sum_{n=1}^N x_n + s \sum_{n=1}^N a_n + \lambda \sum_{n=1}^N p_{best_n}}{(s^2 - (\omega + (N - 1) h) s + \mu) (s^2 + (h - \omega) s + \mu)} \\ &+ \frac{\lambda p_{best_k}}{s (s^2 - (\omega - h) s + \mu)} \\ &+ \frac{\gamma g_{best}}{s (s^2 - (\omega + (N - 1) h) s + \mu)}. \end{aligned} \quad (22)$$

Now it is possible to evaluate the inverse Laplace transform. Firstly, the nonzero poles of (22) are

$$s_{1,2} = \frac{\omega - h \pm \sqrt{(\omega - h)^2 - 4\mu}}{2}, \quad (23)$$

$$s_{3,4} = \frac{\omega + (N - 1) h \pm \sqrt{(\omega + (N - 1) h)^2 - 4\mu}}{2}. \quad (24)$$

If we assume that the parameters satisfy  $(\omega - h)^2 - 4\mu \neq 0$ ,  $(\omega + (N - 1) h)^2 - 4\mu \neq 0$ , all poles will have multiplicity equal to 1. Under this assumption, by applying to (22) the classic residual method for the evaluation of the inverse Laplace transforms, the time domain response is finally obtained:

$$\begin{aligned} x_k(t) &= x_k \sum_{m=1}^2 \frac{(-1)^m s_m e^{s_m t}}{s_2 - s_1} + a_k \sum_{m=1}^2 \frac{(-1)^m e^{s_m t}}{s_2 - s_1} \\ &+ h \sum_{m=1}^4 \frac{\left( \sum_{n=1}^N x_n \right) s_m^2 e^{s_m t}}{\prod_{n=1, n \neq m}^4 (s_m - s_n)} \\ &+ h \sum_{m=1}^4 \frac{\left( \sum_{n=1}^N a_n \right) s_m e^{s_m t}}{\prod_{n=1, n \neq m}^4 (s_m - s_n)} \\ &+ h \sum_{m=1}^4 \frac{\lambda \left( \sum_{n=1}^N p_{best_n} \right) e^{s_m t}}{\prod_{n=1, n \neq m}^4 (s_m - s_n)} \\ &+ \lambda p_{best_k} \left[ \sum_{m=1}^2 \frac{(-1)^m e^{s_m t}}{s_2 - s_1} + \frac{1}{s_1 s_2} \right] \\ &+ \gamma g_{best} \left[ \sum_{m=1}^2 \frac{(-1)^m e^{s_{m+2} t}}{s_{m+2} (s_4 - s_3)} + \frac{1}{s_3 s_4} \right]. \end{aligned} \quad (25)$$

As it has been already said, (25) is valid for a generic  $k$ th particle/bird just within a single time window having width  $\tau$  and starting from  $t_{in}$  (i.e.,  $t_{in} \leq t \leq t_{in} + \tau$ ). It is the closed form which describes how the particle/bird changes its  $j$ th coordinate during time; that is, (25) describes the portion of the trajectory along the  $j$ th dimension followed by the  $k$ th member of the flock within that time window. The corresponding closed form of the velocity along the  $j$ th dimension is trivially obtainable by derivation versus time:

$$\begin{aligned} v_k(t) &= x_k \sum_{m=1}^2 \frac{(-1)^m s_m^2 e^{s_m t}}{s_2 - s_1} + a_k \sum_{m=1}^2 \frac{(-1)^m s_m e^{s_m t}}{s_2 - s_1} \\ &+ h \sum_{m=1}^4 \frac{\left( \sum_{n=1}^N x_n \right) s_m^3 e^{s_m t}}{\prod_{n=1, n \neq m}^4 (s_m - s_n)} \\ &+ h \sum_{m=1}^4 \frac{\left( \sum_{n=1}^N a_n \right) s_m^2 e^{s_m t}}{\prod_{n=1, n \neq m}^4 (s_m - s_n)} \end{aligned}$$

$$\begin{aligned}
 & + h \sum_{m=1}^4 \frac{\lambda \left( \sum_{n=1}^N p_{\text{best}_n} \right) s_m e^{s_m t}}{\prod_{n=1, n \neq m}^4 (s_m - s_n)} \\
 & + \lambda p_{\text{best}_k} \sum_{m=1}^2 \frac{(-1)^m s_m e^{s_m t}}{s_2 - s_1} \\
 & + \gamma g_{\text{best}} \sum_{m=1}^2 \frac{(-1)^m e^{s_{m+2} t}}{(s_4 - s_3)}. \quad (26)
 \end{aligned}$$

These closed forms must be used as updating rules for the position and velocity of each bird at each step/time window of amplitude  $\tau$  variable from a time window to another, just by using  $t = \tau$ . In addition the knowledge of the analytical expression related to the poles allows us to calculate the conditions on CFSO parameters, which provide the full asymptotical stability:

$$\omega < h < -\frac{\omega}{N-1} \cap \omega < 0 \cap \mu > 0. \quad (27)$$

A Matlab implementation (version 1.0) of the previously presented closed forms can be downloaded from the link in [39].

**2.4. Continuous PSO versus CFSO: Effect of the Neighbour-Velocity Matching on the Collective Behaviour.** The stability conditions give us the further possibility to analyze the effect of topological rules on the collective behavior of the swarm/flock members. Firstly, it is worth noticing that we can obtain the close form expressions for a continuous PSO just by posing  $h = 0$  in (25) and (26), obtaining (after some substitutions)

$$\begin{aligned}
 x_k(t) = & x_k \frac{s_2 e^{s_2 t} - s_1 e^{s_1 t}}{\sqrt{\omega^2 - 4\mu}} + a_k \frac{e^{s_2 t} - e^{s_1 t}}{\sqrt{\omega^2 - 4\mu}} \\
 & + (\lambda p_{\text{best}_k} + \gamma g_{\text{best}}) \left( \frac{s_1 e^{s_2 t} - s_2 e^{s_1 t}}{\mu \sqrt{\omega^2 - 4\mu}} + \frac{1}{\mu} \right), \quad (28)
 \end{aligned}$$

$$\begin{aligned}
 v_k(t) = & x_k \frac{s_2^2 e^{s_2 t} - s_1^2 e^{s_1 t}}{\sqrt{\omega^2 - 4\mu}} + a_k \frac{s_2 e^{s_2 t} - s_1 e^{s_1 t}}{\sqrt{\omega^2 - 4\mu}} \\
 & + (\lambda p_{\text{best}_k} + \gamma g_{\text{best}}) \left( \frac{e^{s_2 t} - e^{s_1 t}}{\sqrt{\omega^2 - 4\mu}} \right). \quad (29)
 \end{aligned}$$

As it is possible to observe, for the continuous PSO, the way in which the velocity and the position of a generic  $k$ th particle (28) change during time takes into account the effects of terms which derive from the free response ( $x_k$  and  $v_k$ ) and other terms which derive from the forced response ( $p_{\text{best}_k}$  and  $g_{\text{best}}$ ). Thus, if we consider just the free response of the system, we will have the loss of any contribution coming from other particles (global and personal bests) since the free response of the continuous PSO depends only on the behaviour of the particle itself. On the contrary,

this is not true, for the CFSO, since, in this case, the free response of a generic bird still depends on the behaviour of other members of the flock and the collective behavior exists even if we cancel the contribution of global and personal bests. These observations suggest that the collective behaviour is always present for the CFSO (and consequently for FSO), whereas the CPSO and PSO are rather based on a *forced social* behaviour due to the action of the “global” best.

**2.5. CFSO Algorithm.** The CFSO algorithm can be summarized by the following pseudo-code valid for a generic fitness function,  $f(x^{(1)}, \dots, x^{(D)})$ , to be minimized in the search space  $S \subset \mathbb{R}^D$ , having dimension  $D$ , with  $S \equiv (x^{(1)}, x^{(2)}, \dots, x^{(D)}): x^{(j) \min} \leq x^{(j)} \leq x^{(j) \max}$  for  $j = 1, \dots, D$ .

#### Initialization

Set:

- (i)  $n_{\text{birds}}$  is the total number of elements into the flock;
- (ii) number of birds into the flock controlled by one single bird,  $N_{\text{ctrl\_birds}}$ .
- (iii) Inertial coefficient  $\omega$ , cognitive coefficient,  $\lambda$ , social coefficient,  $\gamma$ , and topological coefficient  $h$  (i.e., choice of the typology of the poles)
- (iv) Maximum number of TWs,  $N_{\text{step\_max}}$ ;
- (v) Velocities for each  $k$ th bird  $v_k^j(t = 0) = \text{random}(-1, 1) \cdot V_{\text{max}}$ ;
- (vi) Position  $(x_k^{(1)}(0), \dots, x_k^{(D)}(0))$  of each  $k$ th bird
- (vii) Initial personal fitness  $f_{p_j}(t = 0)$ ;
- (viii) Initial global fitness  $g(t = 0)$ ;
- (ix) Fitness threshold  $\text{goal\_fitness} = \text{arbitrary small}$ ;

#### Main Loop

For each  $k$ th particle and for TW of amplitude  $\tau$ , until we reach the assigned maximum number of TW,

- (i) update, for each  $k$ th particle, position and velocity at end of considered TW by using (25) and (26) computed for  $t = \tau$ . (let us indicate with  $(x_k^{(1)}, \dots, x_k^{(D)})$  and  $(v_k^{(1)}, \dots, v_k^{(D)})$  respectively the position and the velocity at the end of considered TW)
- (ii) evaluate fitness  $f_k = f(x_k^{(1)}, \dots, x_k^{(D)})$
- (iii) If  $f_k$  is better than the personal best fitness of the  $k$ th particle  $f_{p_k}$ , then assign current position as personal best position and update the personal best fitness:

$$\begin{aligned}
 p_{\text{best}_k}^{(j)} & = x_k^{(j)} \quad \forall j \text{th dimension} \\
 f_{p_k} & = f_k
 \end{aligned} \quad (30)$$

- (iv) If  $f_k$  is better than global best fitness, then assign current position as global best position and update the global best fitness:

$$\begin{aligned} g_{\text{best}}^{(j)} &= x_k^{(j)} \quad \forall j\text{th dimension} \\ g(t) &= f_k \end{aligned} \quad (31)$$

- (v) Assign in new initial conditions  $(x_k^{(1)}, \dots, x_k^{(D)})$ ,  $(v_k^{(1)}, \dots, v_k^{(D)})$  and excitations  $(p_{\text{best}_k}^{(1)}, \dots, p_{\text{best}_k}^{(D)})$  ( $k = 1, \dots, n_{\text{birds}}$ ) and  $(g_{\text{best}}^{(1)}, \dots, g_{\text{best}}^{(D)})$

End for

It is worth noticing again that for each TW the excitation is constant values and from a TW to the successive the continuity of the trajectories is guaranteed by means of the initial conditions, in terms of positions and velocities, evaluated as the last values related to the previous TW.

**2.6. Assignments of Poles for Different Flock Behaviours.** The individuals of a swarm can be driven through three different type of trajectories for three different assignments of the poles: (1) couple of conjugate pure imaginary poles (stable system); (2) poles with real part negative (asymptotically stable system); and (3) poles with real part positive (unstable system). With the aim to investigate deeper on the free response of the CFSO, let us assume the global and all personal bests to be equal to constant values for the whole duration of the simulation. This last assumption allows to investigate the free response since no variation for global as well for personal bests is possible. For the sake of simplicity we refer to a bidimensional space since in this way we have the possibility to plot the trajectories followed by the birds.

**2.6.1. Couple of Conjugate Pure Imaginary Poles.** In this case, by the previous assumption, the forced terms are not allowed to change from a TW to another one; we can look at the trajectory after the end of the transient phase. Since we have imposed a couple of conjugate pure imaginary poles for at least one member of the swarm, the relative components of both position and velocity show a periodic behaviour. Consequently each bird having these kinds of poles will describe (in 2D case) a closed loop as Figure 1 shows. In terms of optimum searching this kind of trajectory can be interpreted as a path made by one member which explores the surrounding space, in order to search for a possible new better solution.

**2.6.2. Poles with Negative Real Part (Asymptotically Stable System).** Clearly the previously described trajectories are not useful from the viewpoint of accuracy in detecting optimum solutions. Thus, in order to refine the values of candidate solutions which could lie in a subspace investigated by members with conjugate pure-imaginary poles, the values of these poles are converted by setting their real part as negative. Indeed, according to (25) and (26), when the transient ends the positions are distributed around the global best and all

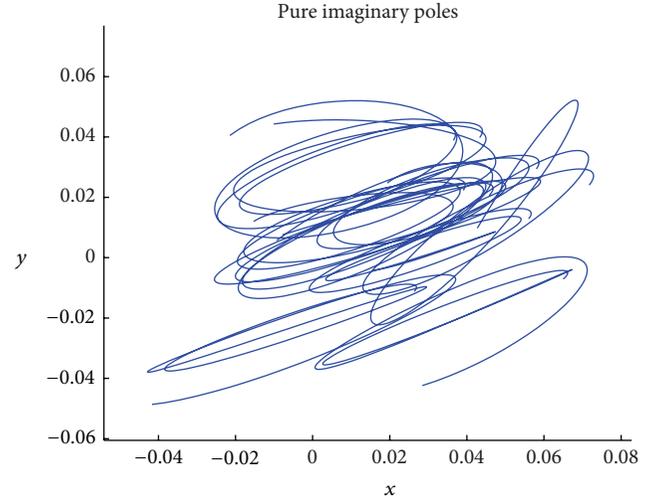


FIGURE 1: Typical trajectories of birds in the flock for pure imaginary poles. It is possible to see the 30 birds starting around the origin, making 30 closed loops.

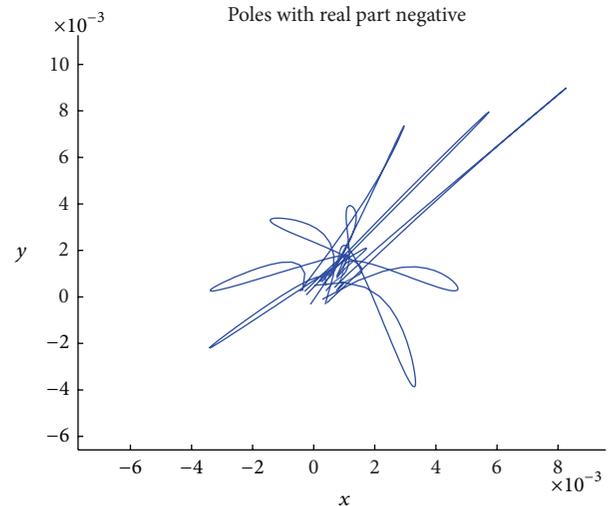


FIGURE 2: Typical trajectories of birds in the flock for asymptotically stable poles. The convergence in the region containing global best is evident that is coincident with the origin of the axes. The effect of refinement can be seen by comparing the values of  $x$  and  $y$  with those of the previous Figure 1.

the velocities have null values. Moreover, during the transient phase, the birds follow convergent trajectories towards the global best (see Figure 2), and consequently a refinement of the solution is obtainable.

**2.6.3. Poles with Positive Real Part (Unstable System).** The last case presented herein is the one concerning poles with positive real part. As previously stated for pure imaginary poles, it is possible that in some situation the birds performing CFSO have fallen in a region containing a local minimum. It is important to have a strategy for escaping from that situation for restarting the exploration. This can be easily done by

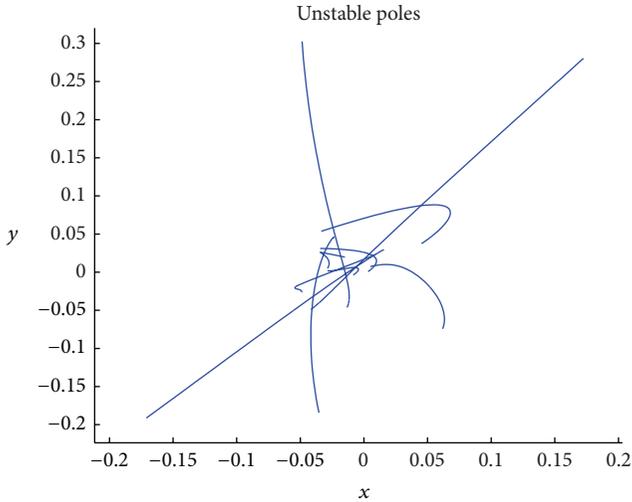


FIGURE 3: Typical trajectories of birds in the flock for unstable poles. The trajectories of 10 birds, starting also in this case around the origin, diverge after few steps, and consequently they escape from minimum.

choosing poles with a positive real part. Indeed, in this case the trajectories will be forced to diverge also in presence of a local minimum (see Figure 3). In addition if the poles have also an imaginary part other than zero, the oscillatory behaviour is combined with the exponential and then they will escape from the entrapping region by following a sort of spiral (unstable focus). Clearly this kind of poles could lead the birds to exit from domain; thus they must be used only for few TWs (eventually just one TW if we choose at the same time a high value for TW amplitude,  $\tau$ , as will be discussed in the next section).

**2.7. Role of the  $\tau$  Parameter.** As previously stated, an important feature of the CFSO is the possibility to predict the behaviour of particles/birds simply studying the poles of Laplace transforms for different TWs. Indeed, by changing these poles we can force the trajectories of the swarm members to converge towards a point, to escape from a possible local minimum, or to stay into a limited zone (closed loops) of the solution space. In this way we can improve the capabilities of exploration and exploitation simply by changing the poles of state equations system at each time window. Thus, by a suitable choice of the poles from a TW to another one, it is possible to control the whole behaviour of the swarm members. On the contrary, the noncontinuous but simply numerical algorithms employ the random updating of parameters to avoid local minima. By using the closed forms of CFSO, the randomness of parameters can be eliminated because the user *a priori* knows how he must manage the parameters to obtain convergence or divergence, that is, how to exalt the exploitation rather than the exploration or *vice versa*. In addition, at any time during the optimization procedure, the CFSO can refine the found solution through a suitable decreasing of the TWs amplitudes, as it will be discussed in the Validation. Indeed, as shown in Figure 4,

it is always possible to refine the solution simply by using small TW amplitudes together with negative real part poles: small TWs return small displacements (consequently a large number of personal bests is investigated), and the refining of values of the candidate solutions is obtainable. In the same way, large TW amplitudes together with positive real part poles are useful to quickly escape from local minima.

### 3. From CFSO to CFSO<sup>3</sup>

On the basis of the previous remarks on CFSO, for enhancing both exploration and convergence, a hybridization strategy similar to the one successfully followed in [40] has been developed. In [40] three different numerical heuristics have been used in such way to exalt their different abilities in exploration, or exploitation or refinement. In the present case, the same idea has been applied to take advantage from the different behaviours provided by CFSO depending on the different setting of its parameters (i.e., the nature of the poles). In particular, at the beginning of the whole optimization process, the CFSO is implemented by setting *pure imaginary poles* (CFSO<sub>pi</sub>). The CFSO<sub>pi</sub> performs exploration and whenever it finds a subregion in which there is a high probability of discovering a global minimum (briefly called “*suspected region*” according to [40]), two operations are made in cascade: (1) the exploitation and the refinement of the solution, by launching a *asymptotically stable* CFSO (CFSO<sub>as</sub>), that is, a CFSO, in which the real part of poles is set to be positive; in such way we can exalt the exploitation of the “*suspected region*” for finding the value of the minimum that lies here (the CFSO population is initialized by means of the result that the CFSO<sub>pi</sub> had just found); (2) the CFSO poles are then temporarily changed in *unstable poles* (CFSO<sub>us</sub>) for  $N$  steps (after which *pure imaginary poles* CFSO is reset), for escaping from the minimum inside the exploited *suspected region* (since it could be a local minimum). This hybridization has been called CFSO<sup>3</sup> in order to underline the three different aspects of CFSO which have been used in synergy. In other words, the optimization process is fully supervised; that is, the change of the nature of poles is *a priori* planned by a criterion based on monitoring the value of the cost function. In the setup of the procedure no stochastic parameter has been used with the exception of the starting positions and velocities of the birds. This means that if one launches CFSO<sup>3</sup> starting from the same initial conditions, it will obtain the same identical results.

**3.1. Parallel Architecture.** CFSO<sup>3</sup> provides its best performances on a distributed architecture, and the algorithm can be fully designed for a master-slave configuration for parallel computation [41, 42]. According to the different CFSO peculiarities related to different setting of the poles, CFSO<sup>3</sup> uses imaginary poles and unstable poles just on the master node, whereas the refinement phase with asymptotically stable poles is performed on slave ones. Virtually, the CFSO<sub>pi</sub> performs exploration of the solution space on the Master node and whenever it finds a “*suspected region*,” two simultaneous operations are made:

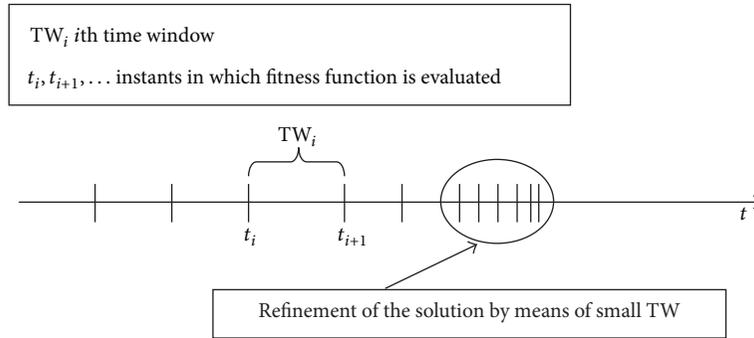


FIGURE 4: Example of managing of time window amplitudes for refining candidate solution.

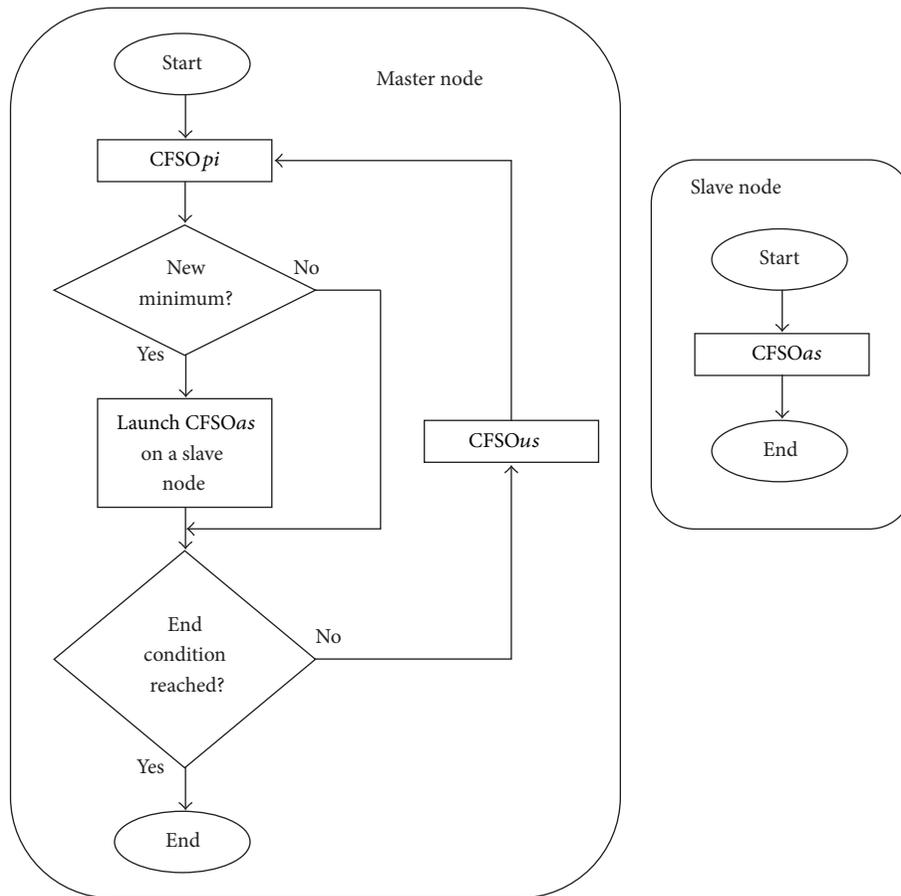
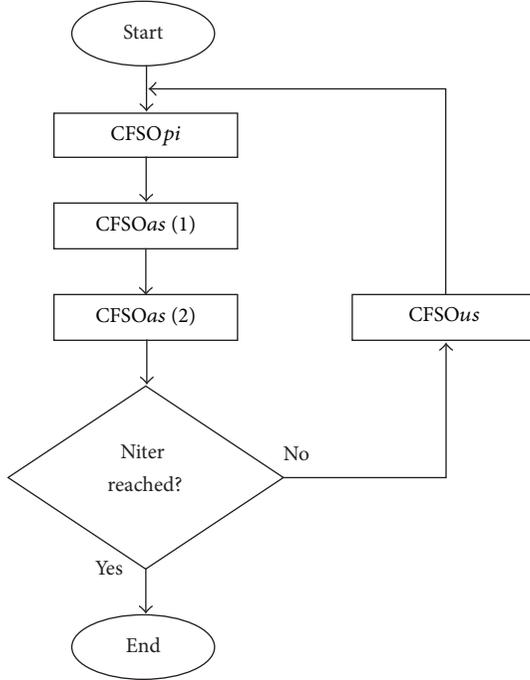


FIGURE 5: Flow chart of parallel version of CFSO<sup>3</sup>.

- (1) CFSO<sub>us</sub> is launched for  $N$  steps (i.e.,  $N$  TWs) on master node (thereafter *imaginary poles* are reset and we return to the CFSO<sub>pi</sub>), in such a way to escape from the “*suspected region*” and to restart exploration;
- (2) CFSO<sub>as</sub> is launched on the first available slave node of the cluster with the aim to refine the solution into that “*suspected region*,” the population of the CFSO<sub>as</sub> with being initialized on the basis of the best result found by CFSO<sub>pi</sub> at the current iteration.

In more detail, the CFSO<sub>as</sub> will be left to run for a fixed number of iterations  $N_{\text{refinement-max}}$  with a duration of TW

(parameter  $\tau$ ) around 10 times smaller than the one used in CFSO<sub>pi</sub> on the master node. While the processes on the slave nodes are running, the CFSO<sub>pi</sub> is provisionally substituted by the CFSO<sub>us</sub> for  $N$  (generally not more than a couple) TWs; thereafter the CFSO<sub>pi</sub> is relaunched. CFSO<sub>pi</sub> still continues to explore the solution space on the Master node and whenever it finds a further “*suspected region*,” another CFSO<sub>as</sub> is launched on a different slave node and so on. After having delivered the found solution to the master, the slave node is ready to be utilized for a further exploration on a new “*suspected region*.” All final results coming from slave nodes are stored in a dedicated file that is located in a shared folder within the master node. Finally, when all processes

FIGURE 6: Flow chart of the CFSO<sup>3</sup> implementation used for tests.

have been concluded (i.e., all PC slaves have completed the refinement procedures previously described, and the CFSO<sub>pi</sub> on the master node has ended all the programmed iterations), a list of all results coming from slave nodes will be available within the master node storage file. Then, the best minimum (global optimum) is trivially identified from that list. The flow chart of the parallel architecture is shown in Figure 5.

#### 4. Validation

The CFSO algorithm has been successfully validated for the solution of engineering design problems [33, 34]. Moreover, it has also been compared with the performances coming from a generic implementation of PSO for the solution of classical optimization benchmarks. It is worth noticing that all validations have been made by starting from the same initial conditions (positions and velocities) [35] with the aim of obtaining the same starting situations as for CFSO, and standard PSO.

Hereafter we follow a validation approach of the CFSO<sup>3</sup> by considering the same tests presented in [35] and other tests on typical benchmarks [43, 44]. In particular, we have used the 2D version of the seven benchmarks proposed in [44]: F1 Shifted Sphere Function, F2 Schwefel's Problem 2.21, F3 Shifted Rosenbrock's Function, F4 Shifted Rastrigin's Function, F5 Shifted Griewank's Function, F6 Shifted Ackley's Function, and the FastFractal "DoubleDip" function, from the fractal function benchmarking suite [44]. The choice of shifted version of benchmarks and the DoubleDip function has the aim to mitigate the effect of possible initialization biases. In particular, we compare the results obtained by CFSO in full connected standard implementation (that was

TABLE 1: Parameters used in CFSO<sup>3</sup>.

	$\omega$	$\lambda$	$\gamma$	$h$	$\tau$
CFSO <sub>pi</sub>	-0.8147	0.421	0.579	9.502e-02	0.2
CFSO <sub>as</sub> (1)	-2.126	0.482	0.913	-0.668	0.05
CFSO <sub>as</sub> (2)	-2.126	0.482	0.913	-0.668	0.02
CFSO <sub>us</sub>	0.126	0.482	0.413	-0.668	0.02

TABLE 2: Success rate (%) of CFSO and CFSO<sup>3</sup> for different number of cycles.

	CFSO (10000 FEs)	CFSO <sup>3</sup> $N_{iter} = 1$ ; (3000 FEs)	CFSO <sup>3</sup> $N_{iter} = 2$ ; (7000 FEs)	CFSO <sup>3</sup> $N_{iter} = 3$ ; (11000 FEs)
Shifted sphere function	30	48	86	97
Schwefel's problem 2.21	36	42	80	98
Shifted Rosenbrock's function	30	46	84	97
Shifted Rastrigin's function	15	24	63	86
Shifted Griewank's function	21	42	81	92
Shifted Ackley's function	23	22	56	70

already successfully compared with PSO in [35]) with those obtainable by the present CFSO<sup>3</sup>, implemented by following the flow chart described in Figure 6, that is, without the use of the parallel architecture. In practice, we repeat  $N_{iter}$  times a cycle in which a CFSO<sub>pi</sub>, two CFSO<sub>as</sub>, and CFSO<sub>us</sub> are sequentially executed, each one for 100 steps and by using the parameters reported in Table 1 (in all cases  $n_{birds} = 10$ ). It is important to note that the deterministic nature of these algorithms causes the identical repetition of trajectories if the same set of guess values is used.

In order to adopt the same initial conditions for all cases, we have used the same positions and velocities as in [35] (100 tests for each benchmark). In particular, we have used 1000 steps for CFSO (the number of steps was the stop criterion adopted), which correspond to an amount of function evaluations (FEs) equal to 10000 for CFSO, whereas for CFSO<sup>3</sup> the number of steps and FEs depend on  $N_{iter}$  according to the expression  $N_{steps} = 300 + 400 \times (N_{iter} - 1)$  and  $FEs = 10 \times N_{steps}$ . The performances for benchmarks (F1-F6) are presented in Table 2 in terms of success rate (% of success), that is, the percentage of tests, for which the error achieved is less than a prefixed threshold (chosen at  $10^{-6}$  for F1, F3, and F5 and  $10^{-4}$  for F2, F4, and F6).

From Table 2 it is possible to see that the success rate increases by increasing the number of cycles executed (success rate around 70% and 97% even after 3 cycles for these examples). Clearly if we use  $N_{iter} = 1$ , practically we make the refinement but we do not use CFSO<sub>us</sub> to escape from possible local minima. However, in this case it is also possible to appreciate an improvement of the performance. These results confirm that, thanks to the refinement approach obtained

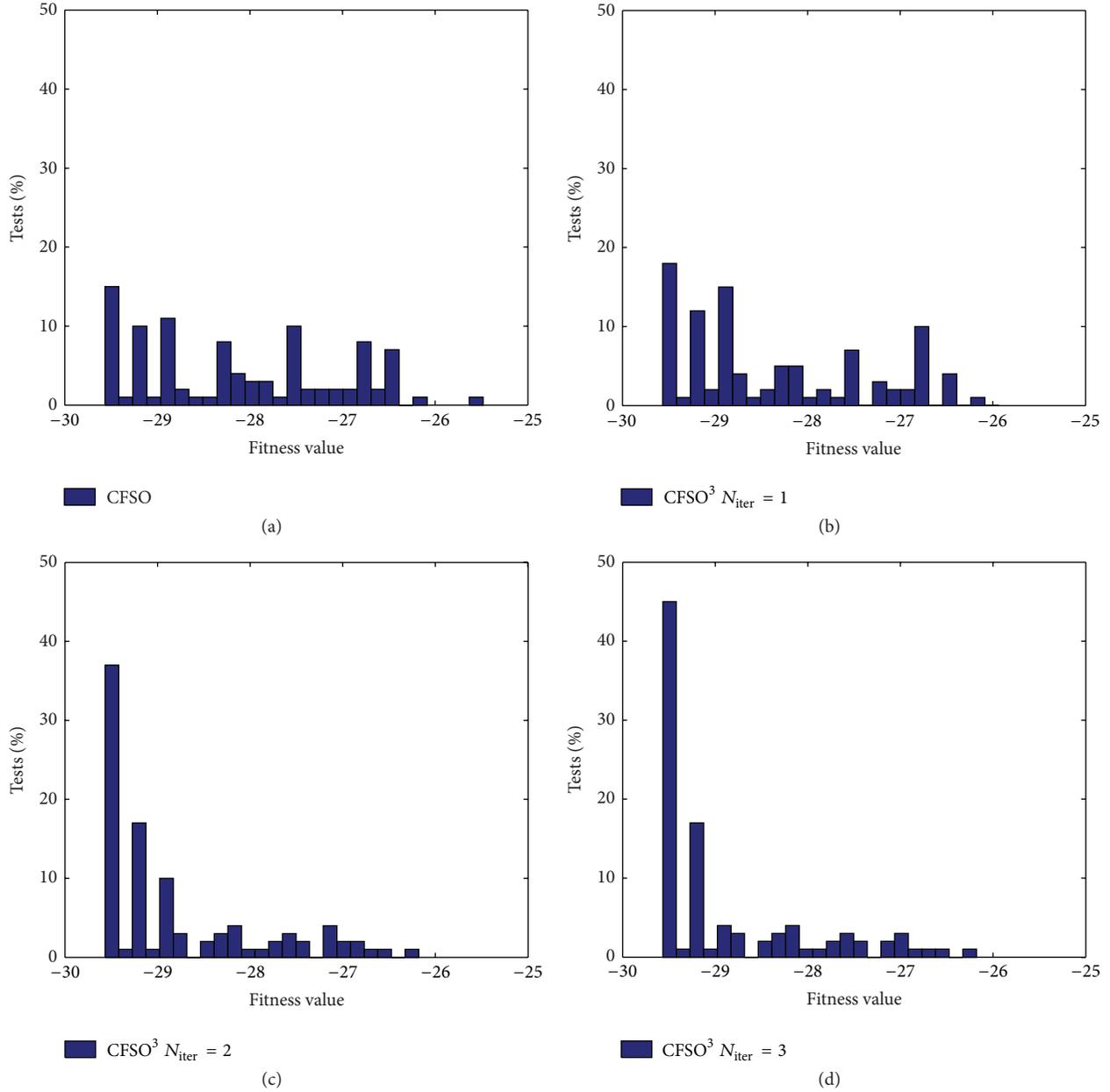


FIGURE 7: The % of tests giving correspondent fitness values for the DoubleDip function.

by CFSoas and to the capability of escaping from local minima given by CFSoUs, we are able to find the solution of an optimization problem almost independently by the initial conditions imposed, that is, without using any stochastic parameters, but rather by a supervised optimization approach. In addition, it is worth noticing that the maximum error achieved after 3 cycles is of few percents, for all the benchmarks.

For the DoubleDip function, since we do not have any information about the values of the global minimum, we present the results by using a set of histograms (shown in Figure 7) representative of the % of tests giving the relative fitness values. As can be seen in Figure 7, the larger the number of iterations, the larger the number of tests giving a minimum percentage value around to 29.56 (the minimum

value obtained among all tests is 29.56615 with respect to 29.5641 obtained in [35]).

Clearly, as previously stated, these results are just early validations, whereas further deeper investigations must be done in order to evaluate the effectiveness of such modification and make the setup of the continuous optimizer more efficient. For example, the nature of the benchmarks F2, F4, and F6, which present a higher gradient, should suggest to increase the number of steps devoted to refinement or to use smaller values for  $\tau$  (in order not to make any change in the code used for the benchmark, the threshold was kept in this case at  $10^{-4}$ ). Nevertheless the promising results obtained in validation confirm the goodness of the proposed supervised deterministic approach. In addition, with the aim of involving the scientific community in the use

and validation of the developed CFSO code, we have provided its MATLAB version 1.0 in [39].

## 5. Conclusions

In this paper a supervised approach based on the hybridization of different CFSO, a heuristics that converts numerical swarm-based algorithms into analytical closed forms, has been presented. The use of the closed forms allows investigating the different types of trajectories that can be performed by swarm members by different parameter setting; that is, what is the influence of the values of parameters on the divergence or the convergence (exploration or exploitation) of the swarm-based algorithms? On the other hand, the development of the CFSO<sup>3</sup> has allowed eliminating the random updating of parameters, a common praxis present in swarm-based algorithms: CFSO<sup>3</sup> manages the parameters switching from a behaviour to another. All these items allow us to design a supervised optimizer, which combines exploration, escaping from minima, and refinement abilities (resp.: CFSO<sub>pi</sub>, CFSO<sub>us</sub>, and CFSO<sub>as</sub>). CFSO<sup>3</sup> can be supported by both sequential and parallel implementations. Although further investigations would be necessary for a deeper analysis of this optimizer, the promising results returned from validations confirm the effectiveness of the proposed approach (a CFSO Matlab code is available in [39]). In addition the CFSO/CFSO<sup>3</sup> equations, defining a continuous dynamic system, can be also hardware implemented by means of an analogical circuit, giving the possibility to put in practice new typology optimizers [34], suitable for the solution of real-time problems also in combination with hybrid artificial intelligence systems which use neural networks [45, 46].

## Appendix

### PSO as Approximation of CPSO Equations

It is interesting to note that the numerical rules of PSO can be obtained by (28) and (29) under some appropriate approximation. Let us start with (28) and rewrite it for  $t = \tau = 1$ :

$$x_k(1) = x_k \frac{s_2 e^{s_2} - s_1 e^{s_1}}{\sqrt{\omega^2 - 4\mu}} + a_k \frac{e^{s_2} - e^{s_1}}{\sqrt{\omega^2 - 4\mu}} + (\lambda p_{\text{best}_k} + \gamma g_{\text{best}}) \left( \frac{s_1 e^{s_2} - s_2 e^{s_1}}{\mu \sqrt{\omega^2 - 4\mu}} + \frac{1}{\mu} \right). \quad (\text{A.1})$$

By substituting the first-order approximation for the exponential

$$e^s \approx 1 + s,$$

$$x_k(1) \approx x_k \frac{s_2(1 + s_2) - s_1(1 + s_1)}{\sqrt{\omega^2 - 4\mu}} + a_k \frac{(1 + s_2) - (1 + s_1)}{\sqrt{\omega^2 - 4\mu}} + (\lambda p_{\text{best}_k} + \gamma g_{\text{best}})$$

$$\begin{aligned} & \times \left( \frac{s_1(1 + s_2) - s_2(1 + s_1)}{\mu \sqrt{\omega^2 - 4\mu}} + \frac{1}{\mu} \right) \\ & = x_k \frac{(s_2 - s_1)(1 + (s_2 + s_1))}{\sqrt{\omega^2 - 4\mu}} + a_k \frac{s_2 - s_1}{\sqrt{\omega^2 - 4\mu}} \\ & \quad + (\lambda p_{\text{best}_k} + \gamma g_{\text{best}}) \left( \frac{s_1 - s_2}{\mu \sqrt{\omega^2 - 4\mu}} + \frac{1}{\mu} \right) \\ & = x_k(1 + (s_2 + s_1)) + a_k = x_k(1 + \omega) + a_k \\ & = x_k + x_k \omega - \omega x_k + u_k = x_k + v_k, \end{aligned} \quad (\text{A.2})$$

that is, the PSO updating rule for positions.

In an analogous way it is possible to retrieve the PSO rule for updating velocity (29):

$$\begin{aligned} v_k(1) & = x_k \frac{s_2^2 e^{s_2} - s_1^2 e^{s_1}}{\sqrt{\omega^2 - 4\mu}} + a_k \frac{s_2 e^{s_2} - s_1 e^{s_1}}{\sqrt{\omega^2 - 4\mu}} \\ & \quad + (\lambda p_{\text{best}_k} + \gamma g_{\text{best}}) \left( \frac{e^{s_2} - e^{s_1}}{\sqrt{\omega^2 - 4\mu}} \right) \\ & \approx x_k \frac{s_2^2(1 + s_2) - s_1^2(1 + s_1)}{\sqrt{\omega^2 - 4\mu}} \\ & \quad + a_k \frac{s_2(1 + s_2) - s_1(1 + s_1)}{\sqrt{\omega^2 - 4\mu}} \\ & \quad + (\lambda p_{\text{best}_k} + \gamma g_{\text{best}}) \left( \frac{1 + s_2 - 1 - s_1}{\sqrt{\omega^2 - 4\mu}} \right) \\ & = \dots \\ & = v_k(1 + \omega) + (\lambda(p_{\text{best}_k} - x_k) + \gamma(g_{\text{best}} - x_k)). \end{aligned} \quad (\text{A.3})$$

From this last equation, it is clear that, in order to obtain the correct numerical version of PSO, an equivalence relation must be established between the inertial parameter  $\omega_{\text{psO}}$  of the PSO algorithm and the inertial parameter  $\omega_{\text{cPSO}}$  of the CPSO:

$$\omega_{\text{psO}} = \omega_{\text{cPSO}} + 1. \quad (\text{A.4})$$

## References

- [1] F. Riganti Fulginei and A. Salvini, "Comparative analysis between modern heuristics and hybrid algorithms," *The International Journal for Computation and Mathematics in Electrical and Electronic Engineering*, vol. 26, no. 2, pp. 259–268, 2007.
- [2] C. W. Liew and M. Labiri, "Exploration or convergence? Another metacontrol mechanism for Gas," in *Proceedings of*

- the 18th International Florida AI Research Society Conference, pp. 251–256, May 2005.
- [3] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of IEEE International Conference on Neural Networks*, vol. 4, pp. 1942–1948, Perth, Australia, December 1995.
  - [4] C. W. Reynolds, "Flocks, herds and schools: a distributed behavioral model," *Computer Graphics*, vol. 21, no. 4, pp. 25–34, 1987.
  - [5] F. Heppner and U. Grenander, "A stochastic nonlinear model for coordinated bird flocks," in *The Ubiquity of Chaos*, S. Krasner, Ed., AAAS Publications, Washington, DC, USA, 1990.
  - [6] A. Engelbrecht, *Fundamentals of Computational Swarm Intelligence*, Wiley, New York, NY, USA, 2005.
  - [7] M. Clerc, *Particle Swarm Optimization*, ISTE, London, UK, 2006.
  - [8] M. R. AlRashidi and M. E. El-Hawary, "A survey of particle swarm optimization applications in electric power systems," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 4, pp. 913–918, 2009.
  - [9] S. Coco, A. Laudani, G. Pollicino, G. Pulcini, F. Riganti Fulginei, and A. Salvini, "TWT magnetic focusing structure optimization by parallel evolutionary algorithm," *The International Journal for Computation and Mathematics in Electrical and Electronic Engineering*, vol. 31, no. 5, pp. 1338–1346, 2012.
  - [10] Y. Shi and R. Eberhart, "A modified particle swarm optimizer," in *Proceedings of IEEE International Conference on Evolutionary Computation (ICEC '98)*, pp. 69–73, May 1998.
  - [11] J. H. Seo, C. H. Im, S. Y. Kwak, C. G. Lee, and H. K. Jung, "An improved particle swarm optimization algorithm mimicking territorial dispute between groups for multimodal function optimization problems," *IEEE Transactions on Magnetics*, vol. 44, no. 6, pp. 1046–1049, 2008.
  - [12] K. Parsopoulos and M. Vrahatis, "Modification of the particle swarm optimizer for locating all the global minima," in *Artificial Neural Networks and Genetic Algorithms*, pp. 324–327, Springer, 2001.
  - [13] D. Bratton and J. Kennedy, "Defining a standard for particle swarm optimization," in *Proceedings of IEEE Swarm Intelligence Symposium (SIS '07)*, pp. 120–127, IEEE Press, April 2007.
  - [14] X. Chen and Y. Li, "A modified PSO structure resulting in high exploration ability with convergence guaranteed," *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 37, no. 5, pp. 1271–1289, 2007.
  - [15] Z. H. Zhan, J. Zhang, Y. Li, and H. S. H. Chung, "Adaptive particle swarm optimization," *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 39, no. 6, pp. 1362–1381, 2009.
  - [16] M. M. Ali and P. Kaelo, "Improved particle swarm algorithms for global optimization," *Applied Mathematics and Computation*, vol. 196, no. 2, pp. 578–593, 2008.
  - [17] M. A. Montes de Oca, T. Stützle, M. Birattari, and M. Dorigo, "Frankenstein's PSO: a composite particle swarm optimization algorithm," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 5, pp. 1120–1132, 2009.
  - [18] Z. Xinchao, "A perturbed particle swarm algorithm for numerical optimization," *Applied Soft Computing*, vol. 10, no. 1, pp. 119–124, 2010.
  - [19] L. Liu, S. Yang, and D. Wang, "Particle swarm optimization with composite particles in dynamic environments," *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 40, no. 6, pp. 1634–1648, 2010.
  - [20] R. Mendes, *Population Topologies and Their Influence in Particle Swarm Performance [Ph.D. thesis]*, Universidade do Minho, Braga, Portugal, 2004.
  - [21] R. Mendes, J. Kennedy, and J. Neves, "The fully informed particle swarm: simpler, maybe better," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, pp. 204–210, 2004.
  - [22] J. Kennedy and R. Mendes, "Neighborhood topologies in fully informed and best-of-neighborhood particle swarms," *IEEE Transactions on Systems, Man, and Cybernetics C*, vol. 36, no. 4, pp. 515–519, 2006.
  - [23] A. S. Mohais, R. Mendes, C. Ward, and C. Posthoff, "Neighborhood re-structuring in particle swarm optimization," in *Proceedings of the 18th Australian Joint Conference on Artificial Intelligence*, vol. 3809 of *Lecture Notes in Computer Science*, p. 776, Sydney, Australia, December 2005.
  - [24] H. Liu, E. Howley, and J. Duggan, "Particle swarm optimization with gradually increasing directed neighbourhoods," in *Proceedings of the 13th Annual Genetic and Evolutionary Computation Conference (GECCO '11)*, pp. 29–36, July 2011.
  - [25] M. A. Montes de Oca, T. Stützle, K. van den Eenden, and M. Dorigo, "Incremental social learning in particle swarms," *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 41, no. 2, pp. 368–384, 2011.
  - [26] X. Li, "Niching without niching parameters: particle swarm optimization using a ring topology," *IEEE Transactions on Evolutionary Computation*, vol. 14, no. 1, pp. 150–169, 2010.
  - [27] F. Riganti Fulginei and A. Salvini, "Hysteresis model identification by the flock-of-starlings optimization," *International Journal of Applied Electromagnetics and Mechanics*, vol. 30, no. 3-4, pp. 321–331, 2009.
  - [28] F. Riganti Fulginei and A. Salvini, "The flock-of starlings optimization: influence of topological rules on the collective behaviour of swarm intelligence," in *Computational Methods for the Innovative Design of Electrical Devices*, Studies in Computational Intelligence, pp. 139–157, Springer, 2010.
  - [29] M. Ballerini, N. Cabibbo, R. Candelier et al., "Interaction ruling animal collective behavior depends on topological rather than metric distance: evidence from a field study," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 105, no. 4, pp. 1232–1237, 2008.
  - [30] J. L. Fernández-Martínez, E. García-Gonzalo, and J. P. Fernández-Alvarez, "Theoretical analysis of particle swarm trajectories through a mechanical analogy," *International Journal of Computational Intelligence Research*, vol. 4, no. 2, pp. 93–104, 2008.
  - [31] H. M. Emara and H. A. Abdelfatah, "Continuous swarm optimization technique with stability analysis," in *Proceedings of the American Control Conference (AAC '04)*, vol. 3, pp. 2811–2817, July 2004.
  - [32] S. M. Mikki and A. A. Kishk, "Physical theory for particle swarm optimization," *Progress in Electromagnetics Research*, vol. 75, pp. 171–207, 2007.
  - [33] S. Coco, A. Laudani, F. Riganti Fulginei, and A. Salvini, "Accurate design of Helmholtz coils for ELF Bioelectromagnetic interaction by means of continuous FSO," *International Journal of Applied Electromagnetics and Mechanics*, vol. 39, no. 1–4, pp. 651–656, 2012.
  - [34] A. Laudani, G. Pulcini, F. Riganti Fulginei, and A. Salvini, "Swarm circuits performing optimization and inverse problems," in *Proceedings of the International Workshops on Optimization and Inverse Problems in Electromagnetism (OIPE '12)*, Ghent, Belgium, September 2012.
  - [35] A. Laudani, F. Riganti Fulginei, and A. Salvini, "Closed forms for the fully-connected continuous flock-of-starlings optimization algorithm," in *Proceedings of the 15th International Conference*

- on *Computer Modeling and Simulation (UKSim '13)*, Cambridge, UK, April 2013.
- [36] W. Li, "Stability analysis of swarms with general topology," *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 38, no. 4, pp. 1084–1097, 2008.
- [37] V. Kadiramanathan, K. Selvarajah, and P. J. Fleming, "Stability analysis of the particle dynamics in particle swarm optimizer," *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 3, pp. 245–255, 2006.
- [38] S. Kiranyaz, T. Ince, A. Yildirim, and M. Gabbouj, "Fractional particle swarm optimization in multidimensional search space," *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 40, no. 2, pp. 298–319, 2010.
- [39] CFSO—Continuous Flock-of-Starlings Optimization Algorithm tool, 2013, <http://www.dea.uniroma3.it/elettrotecnica/>.
- [40] F. Riganti Fulginei, A. Salvini, and G. Pulcini, "Metric-topological-evolutionary optimization," *Inverse Problems in Science and Engineering*, vol. 20, no. 1, pp. 41–58, 2012.
- [41] S. Coco, A. Laudani, G. Pulcini, F. Riganti Fulginei, and A. Salvini, "Shape optimization of multistage depressed collectors by parallel evolutionary algorithm," *IEEE Transactions on Magnetics*, vol. 48, no. 2, pp. 435–438, 2012.
- [42] S. Coco, A. Laudani, F. Riganti Fulginei, and A. Salvini, "TEAM problem 22 approached by a hybrid artificial life method," *The International Journal for Computation and Mathematics in Electrical and Electronic Engineering*, vol. 31, no. 3, pp. 816–826, 2012.
- [43] C. MacNish, "Towards unbiased benchmarking of evolutionary and hybrid algorithms for real-valued optimisation," *Connection Science*, vol. 19, no. 4, pp. 361–385, 2007.
- [44] K. Tang, X. Yao, P. N. Suganthan et al., "Benchmark functions for the CEC'2008 special session and competition on large scale global optimization," Tech. Rep., Nature Inspired Computation and Applications Laboratory, USTC, China, 2007.
- [45] M. Gneo, M. Schmid, S. Conforto, and T. D'Alessio, "A free geometry model-independent neural eye-gaze tracking system," *Journal of NeuroEngineering and Rehabilitation*, vol. 9, article 82, 2012.
- [46] I. Bernabucci, S. Conforto, M. Capozza, N. Accornero, M. Schmid, and T. D'Alessio, "A biologically inspired neural network controller for ballistic arm movements," *Journal of NeuroEngineering and Rehabilitation*, vol. 4, article 33, 2007.

## Research Article

# Weighted-Bit-Flipping-Based Sequential Scheduling Decoding Algorithms for LDPC Codes

**Qing Zhu and Le-nan Wu**

*School of Information Science and Engineering, Southeast University, Nanjing 210096, China*

Correspondence should be addressed to Qing Zhu; [dreamathstat@163.com](mailto:dreamathstat@163.com)

Received 24 April 2013; Accepted 22 June 2013

Academic Editor: Yudong Zhang

Copyright © 2013 Q. Zhu and L.-n. Wu. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Low-density parity-check (LDPC) codes can be applied in a lot of different scenarios such as video broadcasting and satellite communications. LDPC codes are commonly decoded by an iterative algorithm called belief propagation (BP) over the corresponding Tanner graph. The original BP updates all the variable-nodes simultaneously, followed by all the check-nodes simultaneously as well. We propose a sequential scheduling algorithm based on weighted bit-flipping (WBF) algorithm for the sake of improving the convergence speed. Notoriously, WBF is a low-complexity and simple algorithm. We combine it with BP to obtain advantages of these two algorithms. Flipping function used in WBF is borrowed to determine the priority of scheduling. Simulation results show that it can provide a good tradeoff between FER performance and computation complexity for short-length LDPC codes.

## 1. Introduction

Low-density parity-check (LDPC) codes were first invented by Gallager [1] but had been neglected for decades until Mackay brought them back to light in 1996 [2]. Since then, much attention had been attracted for their excellent Shannon limit approaching error-correcting performance through belief propagation (BP) [3] decoding algorithm. This iterative decoding algorithm, sometimes also called sum-product algorithm (SPA) [4], is a powerful algorithm to approximately solve many NP hard problems such as statistical inference in physics [5–7], hypothesis testing, cooperative localization, and channel coding.

There are lots of researches with various decoding algorithms. Among existing LDPC decoding algorithms, bit-flipping (BF) algorithms are the simplest. The operations of check-nodes in BF are modulo-two additions while the variable-nodes only need simple comparison operations. BF decoding algorithms are easy to implement, but they usually perform not so well when compared to BP decoding algorithms, so various weighted BF (WBF) [8–12] decoding algorithms were proposed.

To improve the standard BP decoding performance, several sequential scheduling strategies in BP have been invented. In sequential scheduling strategies, the messages are computed in a serial manner using the newest updated information. Sequential strategies were introduced as a sequence updates based on check-node (CSBP) [13–15] or variable-node (VSBP) [16, 17]. Simulations demonstrate that sequential strategies converge about twice as fast as the standard parallel BP decoding algorithms (Flood) without any extra computing burden. The kernel steps of sequential updating algorithms focus on finding the order of message updating which converges fastest. To our knowledge, the best decoding algorithms in the sense of performance is informed dynamic scheduling (IDS) [18–24] algorithms, which update messages dynamically. A metric called residual [18] is used in IDS which decides the updating order of propagated messages. Metric computing and selecting operations can cause significant increase in computational complexity in IDS. In order to achieve the tradeoff between decoding performance and computational complexity, a low-complexity sequential WBF-based scheduling algorithm is proposed, in which the priority used in WBF determines the order of scheduling. Simulation results show that it can provide a good tradeoff

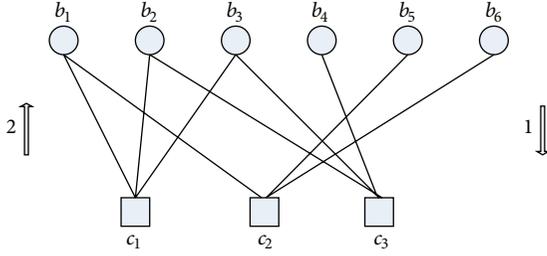


FIGURE 1: Standard BP algorithm (Flood).

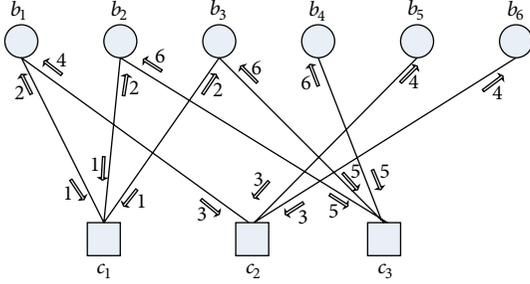


FIGURE 2: CSBP algorithm.

TABLE 1: Computation complexity from variable-nodes to check-nodes.

Algorithms	Updating	Reliability computation	Sorting
Flood	$e$	0	0
CSBP	$e$	0	0
IDS	$Md_c(d_v - 1)$	0	0
WBFSBP	$e$	0	0

TABLE 2: Computation complexity from check-nodes to variable-nodes.

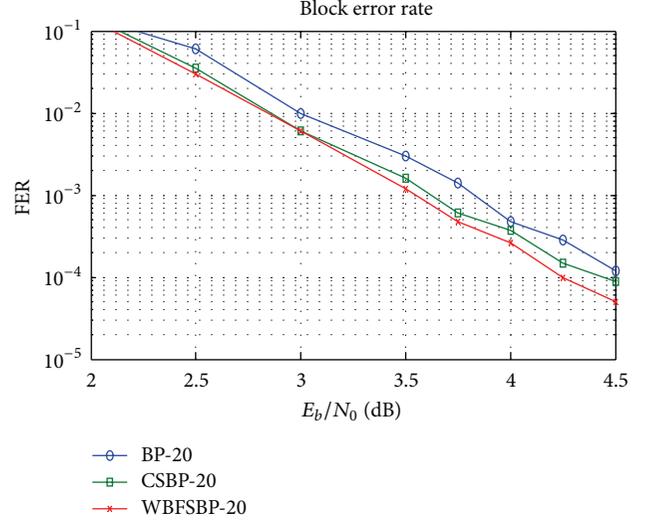
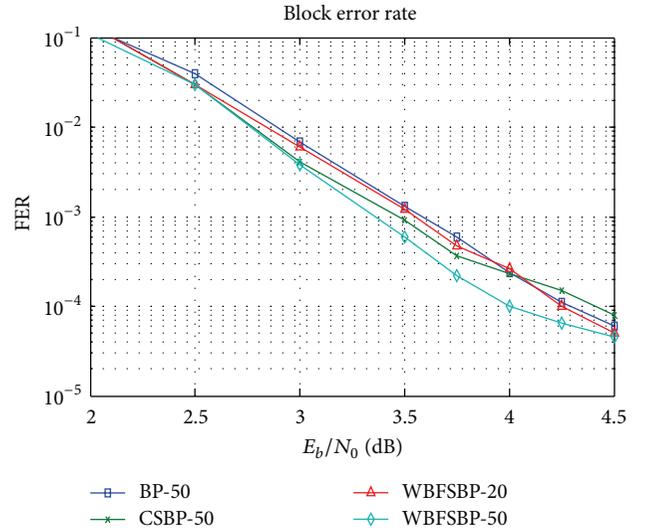
Algorithms	Updating	Reliability computation	Sorting
Flood	$e$	0	0
CSBP	$e$	0	0
IDS	$e$	$Md_c(d_v - 1)(d_c - 1)$	$\mathcal{O}(N^2)$
WBFSBP	$e$	$Md_c d_v$	$\mathcal{O}(N \log N)$

between block errors performance and complexity for short-length LDPC codes.

This paper is organized as follows. Section 2 reviews BP (Flood and CSBP) and WBF. Section 3 introduces our WBF-based serial BP (WBFSBP) strategies. Section 4 analyzes the computing complexity of WBFSBP. Section 5 reveals the simulation results. Section 6 draws the conclusions.

## 2. WBF and BP

Let  $N$  and  $K$  be the block length and the information length of a binary LDPC code; thus the rate of the code is  $r = K/N$ . Let  $M = N - K$ ; an LDPC code is described by an  $M \times N$  parity-check matrix  $\mathbf{H}$ .  $H_{mn}$  presents the entry of row  $m$  and column  $n$  in  $\mathbf{H}$ . The set  $\mathcal{N}(m)$  denotes the nodes that adjoin

FIGURE 3: FER versus  $E_b/N_0$  performance of BP (Flood), CSBP, and WBFSBP decoding algorithms with maximal number of iterations of 20 with LDPC code of blocklength of 256 and rate of 0.5. BP algorithm with maximal number of iterations of 20.FIGURE 4: FER versus  $E_b/N_0$  performance of BP (Flood), CSBP and WBFSBP decoding algorithms with maximal number of iterations of 50 with LDPC code of blocklength of 256 and rate of 0.5. As compared, performance of max iteration 20 of WBFSBP is given.

$m$  by  $\mathcal{N}(m) = \{n : H_{mn} = 1\}$ , similarly for the set  $\mathcal{N}(n) = \{m : H_{mn} = 1\}$ .  $\mathcal{N}(m)/n$  denotes the set  $\mathcal{N}(m)$  excluding  $n$ . Binary phase shift keying (BPSK) modulation over an additive white Gauss noise (AWGN) channel is assumed for information transmission, which maps a codeword vector  $\mathbf{c} = (c_1, c_2, \dots, c_n)$  into a vector  $\mathbf{x} = (x_1, x_2, \dots, x_n)$ , where  $c_i = 2x_i - 1$ ,  $i \in [1, N]$ . The received vector is  $\mathbf{y} = (y_1, y_2, \dots, y_n)$ , where  $y_i = x_i + n_i$ ,  $i \in [1, N]$ , and  $n_i$  is the AWGN.  $\mathbf{Z} = (z_1, z_2, \dots, z_n)$  denotes the hard-decision vector acquired from  $\mathbf{y}$ :  $z_i = 1$  if  $y_i > 0$  and  $z_i = 0$  if  $y_i \leq 0$ .

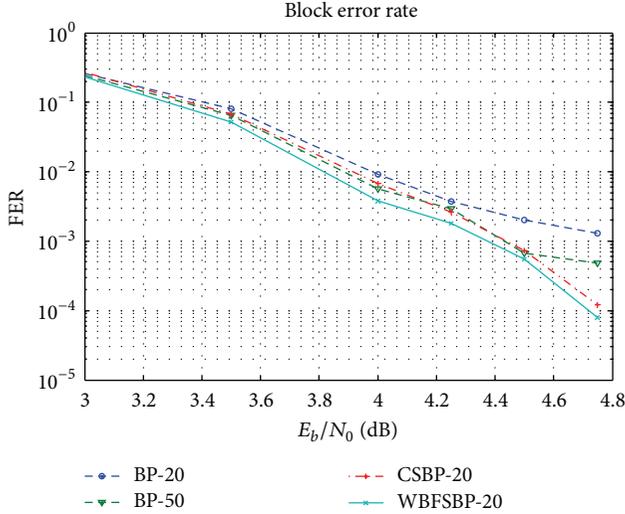


FIGURE 5: FER versus  $E_b/N_0$  performance of BP (Flood), CSBP and WBFSBP decoding algorithms with maximal number of iterations of 20 with LDPC code of blocklength of 204 and rate of 0.5. BP-20 denotes BP algorithm with maximal number of iterations of 20. As compared, performance of max iteration 50 of BP is given.

$\mathbf{S} = (s_1, s_2, \dots, s_M) = \mathbf{z} \cdot H^T$  denotes the syndrome of hard-decision vector  $\mathbf{z}$ .

**2.1. WBF.** Firstly, a metric called “reliability” and a function-associated flipping probability of each check-node are calculated via [9]

$$w_{n,m} = \min_{i \in \mathcal{N}(m) \setminus n} |y_i|, \quad m \in [1, M], \quad n \in \mathcal{N}(m),$$

$$e_{\text{WBF},n} = \sum_{m \in \mathcal{N}(n)} (2s_m - 1) w_{n,m} - \alpha |y_n|. \quad (1)$$

The WBF algorithm is described as follows [9]:

- (1) set iteration number  $k = 0$ , take  $k_{\max}$  as the maximum number of iterations, and pick up  $w_{n,m}$ ,
- (2) calculate  $s^k = \mathbf{z}^k H^T$ ; if  $s^k = 0$ , stop decoding and output  $\mathbf{z}^k$ ,
- (3) for  $n = 1, 2, \dots, N$ , calculate  $e_{\text{WBF},n}^k$  as follows:

$$e_{\text{WBF},n}^k = \sum_{m \in \mathcal{N}(n)} (2s_m^k - 1) w_{n,m} - \alpha |y_n|, \quad (2)$$

- (4)  $z^{n^*} = z^{n^*} + 1$  based on

$$n^* = \arg \max_{n \in [1, N]} e_{\text{WBF},n}^k, \quad (3)$$

- (5)  $k = k + 1$  and go to step (3), until stopping rules are satisfied.

**2.2. Standard BP (Flood) Decoding Algorithm.** Commonly, the communication of the nodes in the corresponding Tanner graph comprises the Flood algorithm. For every variable-node  $v_j$  and check-node  $c_i$ , the corresponding message generation functions  $m_{c_i \rightarrow v_j} = f_{c_i \rightarrow v_j}(m)$  and  $m_{v_j \rightarrow c_i} = f_{v_j \rightarrow c_i}(m)$  are defined as follows [2]:

$$m_{v_j \rightarrow c_i} = \sum_{c_a \in \mathcal{N}(v_j) \setminus c_i} m_{c_a \rightarrow v_j} + C_{v_j},$$

$$m_{c_i \rightarrow v_j} = 2 \operatorname{arctanh} \left( \prod_{v_b \in \mathcal{N}(c_i) \setminus v_j} \tanh \left( \frac{m_{v_b \rightarrow c_i}}{2} \right) \right), \quad (4)$$

$$L(v_j) = \sum_{c_a \in \mathcal{N}(v_j)} m_{c_a \rightarrow v_j} + C_{v_j},$$

where  $C_{v_j} = \log(p(y_j | v_j = 0)/p(y_j | v_j = 1))$  is the channel information of  $v_j$ .  $L(v_j)$  denotes the LLR of  $v_j$ .

As shown in Figure 1, in time slot 1, messages are passed to all of these check-nodes by all variable-nodes that connect to them, and in time slot 2, all of the processed messages are sent back to all variable-nodes. The Flood decoding algorithm is formally described in Pseudocode 1.

**2.3. CSBP.** The messages calculated in CSBP [15] are the same as those in the Flood decoding algorithms. The only difference between them is the way of messages updating. In every iteration of CSBP, the messages are updated in a sequential way: update every check-node one by one in an ascending order (or descending order) arranged in the corresponding Tanner graph. The passing process of messages in CSBP is shown in Figure 2. Check-nodes are represented by  $(c_1, c_2, c_3)$ , and  $(b_1, b_2, \dots, b_6)$  for variable-nodes. In time slot 1, messages are passed to  $c_1$  by all variable-nodes that connect to  $c_1$ . In time slot 2, the processed messages are sent back to those variable-nodes. The updating rules of  $c_2$  and  $c_3$  are identical with those of  $c_1$ . The realization of CSBP is described in Pseudocode 2.

### 3. WBF-Based Serial BP

We will introduce our WBFSBP algorithm using a new message updating schedule. In general, sequential decoding algorithms are composed of two major steps.

*Step 1.* Determine the order of nodes to be updated.

*Step 2.* Compute the messages.

In Flood and CSBP, Step 1 is skipped. In IDS, these two steps are operated alternately for every node. It means that after updating each single node, “residual” computing and sorting operation is needed, which causes much extra computing complexity. For the sake of getting the tradeoff between performance and computing complexity, a simple sequential decoding algorithm called WBF-based serial BP is proposed. First, let us review the flipping function ( $e_{\text{WBF},n}^k$ )

```

1: Set every  $m_{c \rightarrow v} = 0$ 
2: Set every  $m_{v_j \rightarrow c_i} = C_{v_j}$ 
3: while stop rules unsatisfied do
4:   for all  $c_a$  ( $a = 1, 2, \dots, M$ )
5:     for each  $v_b \in \mathcal{N}(c_a)$ , compute and generate  $m_{v_b \rightarrow c_a}$  ( $a = 1, 2, \dots, M$ ) simultaneously
6:     for each  $v_b \in \mathcal{N}(c_a)$ , compute and generate  $m_{c_a \rightarrow v_b}$  ( $a = 1, 2, \dots, M$ ) simultaneously
7: end while

```

PSEUDOCODE 1: Pseudocodes of Flood.

```

1: Set every  $m_{c \rightarrow v} = 0$ 
2: Set every  $m_{v_j \rightarrow c_i} = C_{v_j}$ 
3: while stop rules unsatisfied do
4:   for  $i$  do
5:     for check-node  $c_i$ 
6:       for every  $v_b \in \mathcal{N}(c_i)$  do
7:         compute and generate  $m_{v_b \rightarrow c_i}$ 
8:       end for
9:       for every  $v_b \in \mathcal{N}(c_i)$  do
10:        compute and generate  $m_{c_i \rightarrow v_b}$ 
11:      end for
12:    end for
13: end while

```

PSEUDOCODE 2: Pseudocodes of CSBP.

of a variable-node in WBF decoding algorithm described in Section 2. The larger the  $e_{\text{WBF},n}^k$  is, the more unreliable the corresponding variable-node is. For each check-node, a new metric is defined as follows:

$$E_{\text{WBF},m}^k = \max_{n \in \mathcal{N}(m)} e_{\text{WBF},n}^k - \text{sub} \max_{n \in \mathcal{N}(m)} e_{\text{WBF},n}^k, \quad (5)$$

where  $\text{submax}(\cdot)$  denotes the second maximal value of  $(\cdot)$ . If there is more than one unreliable variable-node connected to a check-node, the message passing within this check-node is almost nonsense. If all neighbor variable-nodes of a check-node are reliable, the priority of updating this check-node should be very low. So we concentrate on the check-node which has only one unreliable neighbor variable-node. The check-nodes with larger value of  $E_{\text{WBF},m}^k$  are more likely containing just one unreliable neighbor variable-node. So updating these check-nodes firstly can correct errors in time and speed up convergence, thus reducing the iterations. In contrast to IDS, the sorting operation is not dynamically decided node by node but is simply obtained from sorting the set of  $E_{\text{WBF},m}^k$  in each iteration. When the sorting is finished, the rest of updating computation is the same as that of CSBP. The detailed steps of WBFSBP are shown in Pseudocode 3.

#### 4. Complexity

Let  $d_v$  and  $d_c$  represent the degree of any variable-node and check-node, respectively;  $e$  is the total number of all edges in the corresponding Tanner graph, so  $e = d_v \cdot N = d_c \cdot$

$M$ , where  $N$  is the number of variable-nodes and  $M$  is the number of check-nodes. In Tables 1 and 2, we have listed the computation complexity of check-to-variable phase and variable-to-check phase, respectively. In an iteration of Flood, every edge of the Tanner graph should be updated once for each direction, so the number of updated messages via check-to-variable and variable-to-check is  $e$ , respectively. Sequential updating strategies surpass the Flood without any extra cost. We note that the check-to-variable message updating computations in one IDS iteration is the same as reverse message updating computations in one WBFSBP iteration, both equal  $e$ . In each iteration, sorting of  $\{E_{\text{WBF},m}^k\}$  requires  $\mathcal{O}(N \log N)$  operations while dynamic scheduling strategies in IDS need  $\mathcal{O}(N^2)$ .

#### 5. Simulation Results

The decoding performance of the Flood, CSBP, and WBFSBP over AWGN channels is presented in this part. The LDPC codes are constructed based on Gallager's random method without any 4-cycle.

Figures 3 and 4 present the FER performance of the (128, 256) Gallager LDPC code with BP (Flood), CSBP, and WBFSBP decoding algorithms. The  $k_{\text{max}}$  is set to 20 and 50, respectively, and  $\alpha$  is set to 1.3 in the computation of  $E_{\text{WBF},m}^k$ . Figure 3 shows that, at the FER of  $1e - 4$ , WBFSBP acquires about 0.2 dB and 0.35 dB promotions over the CSBP and BP in the case of  $k_{\text{max}} = 20$ , while in Figure 4, the coding gains are 0.25 dB and 0.45 dB in the case of  $k_{\text{max}} = 50$ . Figure 4 also

```

0: Set iteration number  $k = 0$ , take  $k_{\max}$  as the maximum number of iteration, pick up  $w_{n,m}$ ,
   Set every  $m_{c \rightarrow v} = 0$ , every  $m_{v_j \rightarrow c_i} = C_{v_j}$ .
1: Compute  $s^k = z^k H^T$ , if  $s^k = \mathbf{0}$ , stop decoding and output  $z^k$ .
2: For  $n = 1, 2, \dots, N$ , compute

$$e_{\text{WBF},n}^k = \sum_{m \in \mathcal{N}(n)} (2s_m^k - 1)w_{n,m} - \alpha |y_n|$$

3: sort  $E_{\text{WBF},m}^k$  in descending order and put the corresponding check-nodes to  $U$ 
4: while stop rules unsatisfied do
5:   for  $i$  do
6:     for  $i$ th check-node  $c_{U_i}$  in  $U$ 
7:       for every  $v_b \in \mathcal{N}(c_{U_i})$  do
8:         compute and generate  $m_{v_b \rightarrow c_{U_i}}$ 
9:       end for
10:      for every  $v_b \in \mathcal{N}(c_{U_i})$  do
11:        compute and generate  $m_{c_{U_i} \rightarrow v_b}$ 
12:      end for
13:    end for
14:  end while

```

PSEUDOCODE 3: Pseudocodes of WBFSBP.

shows that at  $E_b/N_0 = 4$ , WBFSBP-20 can achieve the same FER performance as what BP-50 and CSBP-50 do. In other words, WBFSBP algorithm can reduce more than half of the iterations of that BP algorithm needed at some  $E_b/N_0$ .

Figure 5 presents the FER performance of the (102, 204) Gallager LDPC code with above-mentioned decoding algorithms. The  $k_{\max}$  is set to 20 and  $\alpha$  is set to 1.3. We see that WBFSBP-20 can beat BP-50 for all  $E_b/N_0$ . Compared to CSBP-20, WBFSBP-20 can acquire about 0.15 dB coding gain.

## 6. Conclusion

In this paper, a new sequential scheduling decoding algorithm is proposed, in which the order of message passing is based on the weight factor computed in WBF algorithm. We focus on the check-nodes which have just one unreliable neighbor variable-node. Updating these check-nodes preferentially can avoid errors of propagation, thus speeding up the convergence of decoding algorithm. Compared to IDS, our decoding algorithm gives a good tradeoff between error performance and decoding complexity. As a future work, the scheduling algorithm can be generalized to LDPC codes over GF( $q$ ).

## Acknowledgments

The authors would like to thank all anonymous reviewers whose constructive suggestions were very helpful. The work of this paper is sponsored by the State 863 Project (2008AA01Z227), the National Natural Science Foundation of China (NSFC), under Grant 61271204.

## References

- [1] R. G. Gallager, "Low-density parity-check codes," *IRE Transactions on Information Theory*, vol. 8, no. 1, pp. 21–28, 1962.

- [2] D. J. C. MacKay and R. M. Neal, "Near Shannon limit performance of low density parity check codes," *Electronics Letters*, vol. 32, no. 18, pp. 1645–1646, 1996.
- [3] R. J. McEliece, D. J. C. MacKay, and J.-F. Cheng, "Turbo decoding as an instance of Pearl's "belief propagation" algorithm," *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 2, pp. 140–152, 1998.
- [4] F. R. Kschischang, B. J. R. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 498–519, 2001.
- [5] Y. Zhang, Y. Jun, G. Wei, and L. Wu, "Find multi-objective paths in stochastic networks via chaotic immune PSO," *Expert Systems with Applications*, vol. 37, no. 3, pp. 1911–1919, 2010.
- [6] Y. Zhang, L. Wu, G. Wei, and S. Wang, "A novel algorithm for all pairs shortest path problem based on matrix multiplication and pulse coupled neural network," *Digital Signal Processing*, vol. 21, no. 4, pp. 517–521, 2011.
- [7] Y. Zhang and L. Wu, "Optimal multi-level thresholding based on maximum Tsallis entropy via an artificial bee colony approach," *Entropy*, vol. 13, no. 4, pp. 841–859, 2011.
- [8] D. Burshtein, "On the error correction of regular LDPC codes using the flipping algorithm," *IEEE Transactions on Information Theory*, vol. 54, no. 2, pp. 517–530, 2008.
- [9] M. Jiang, C. Zhao, Z. Shi, and Y. Chen, "An improvement on the modified weighted bit flipping decoding algorithm for LDPC codes," *IEEE Communications Letters*, vol. 9, no. 9, pp. 814–816, 2005.
- [10] X. Wu, C. Zhao, and X. You, "Parallel weighted bit-flipping decoding," *IEEE Communications Letters*, vol. 11, no. 8, pp. 671–673, 2007.
- [11] X. Wu, C. Ling, M. Jiang, E. Xu, C. Zhao, and X. You, "New insights into weighted bit-flipping decoding," *IEEE Transactions on Communications*, vol. 57, no. 8, pp. 2177–2180, 2009.
- [12] N. Miladinovic and M. P. C. Fossorier, "Improved bit-flipping decoding of low-density parity-check codes," *IEEE Transactions on Information Theory*, vol. 51, no. 4, pp. 1594–1606, 2005.

- [13] H. Kfir and I. Kanter, "Parallel versus sequential updating for belief propagation decoding," *Physica A*, vol. 330, no. 1-2, pp. 259–270, 2003.
- [14] M. M. Mansour and N. R. Shanbhag, "High-throughput LDPC decoders," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 11, no. 6, pp. 976–996, 2003.
- [15] D. E. Hocevar, "A reduced complexity decoder architecture via layered decoding of LDPC codes," in *Proceedings of the IEEE Workshop on Signal Processing Systems Design and Implementation (SIPS '04)*, pp. 107–112, October 2004.
- [16] J. Zhang, Y. Wang, M. Fossorier, and J. S. Yedidia, "Replica shuffled iterative decoding," in *Proceedings of the IEEE International Symposium on Information Theory Proceedings (ISIT '05)*, pp. 454–458, 2005.
- [17] J. Zhang and M. P. C. Fossorier, "Shuffled iterative decoding," *IEEE Transactions on Communications*, vol. 53, no. 2, pp. 209–213, 2005.
- [18] A. I. V. Casado, M. Griot, and R. D. Wesel, "Improving LDPC decoders via informed dynamic scheduling," in *Proceedings of the IEEE Information Theory Workshop (ITW '07)*, pp. 208–213, Tahoe City, CA, USA, September 2007.
- [19] A. I. V. Casado, M. Griot, and R. D. Wesel, "Informed dynamic scheduling for belief-propagation decoding of LDPC codes," in *Proceedings of the IEEE International Conference on Communications (ICC '07)*, pp. 932–937, Glasgow, UK, June 2007.
- [20] D. Levin, S. Litsyn, and E. Sharon, "Lazy scheduling for LDPC decoding," *IEEE Communications Letters*, vol. 11, no. 1, pp. 70–72, 2007.
- [21] A. I. V. Casado, M. Griot, and R. D. Wesel, "LDPC decoders with informed dynamic scheduling," *IEEE Transactions on Communications*, vol. 58, no. 12, pp. 3470–3479, 2010.
- [22] J.-H. Kim, M.-Y. Nam, and H.-Y. Song, "Variable-to-check residual belief propagation for LDPC codes," *Electronics Letters*, vol. 45, no. 2, pp. 117–118, 2009.
- [23] G. J. Han and X. Liu, "An efficient dynamic schedule for layered belief-propagation decoding of LDPC codes," *IEEE Communications Letters*, vol. 13, no. 12, pp. 950–952, 2009.
- [24] Y. Gong, X. Liu, W. Ye, and G. J. Han, "Effective informed dynamic scheduling for belief propagation decoding of LDPC codes," *IEEE Transactions on Communications*, vol. 59, no. 10, pp. 2683–2691, 2011.

## Research Article

# Artificial Bee Colony Algorithm Merged with Pheromone Communication Mechanism for the 0-1 Multidimensional Knapsack Problem

Junzhong Ji, Hongkai Wei, Chunnian Liu, and Baocai Yin

College of Computer Science and Technology, Beijing University of Technology, Beijing Municipal Key Laboratory of Multimedia and Intelligent Software Technology, Beijing 100124, China

Correspondence should be addressed to Junzhong Ji; jjz01@bjut.edu.cn

Received 24 March 2013; Accepted 11 June 2013

Academic Editor: Vishal Bhatnagar

Copyright © 2013 Junzhong Ji et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Given a set of  $n$  objects, the objective of the 0-1 multidimensional knapsack problem (MKP\_01) is to find a subset of the object set that maximizes the total profit of the objects in the subset while satisfying  $m$  knapsack constraints. In this paper, we have proposed a new artificial bee colony (ABC) algorithm for the MKP\_01. The new ABC algorithm introduces a novel communication mechanism among bees, which bases on the updating and diffusion of inductive pheromone produced by bees. In a number of experiments and comparisons, our approach obtains better quality solutions in shorter time than the ABC algorithm without the mechanism. We have also compared the solution performance of our approach against some stochastic approaches recently reported in the literature. Computational results demonstrate the superiority of the new ABC approach over all the other approaches.

## 1. Introduction

Given a set  $J = \{o_1, o_2, \dots, o_n\}$  of  $n$  objects and a knapsack with a set  $C = \{c_1, c_2, \dots, c_m\}$  of  $m$  dimensions, the 0-1 multidimensional knapsack problem (MKP\_01) seeks a subset of  $J$  in such a way that the total profit of objects included in the subset is maximized, while  $m$  resource constraints remain satisfied. More formally, each object  $o_j \in J$  has profit  $p_j$  and weight  $r_{ij}$  in dimension  $i$  ( $1 \leq i \leq m$ ), and each dimension of the knapsack has a capacity  $c_i$ . By introducing binary decision variable  $x_j$  to indicate whether object  $o_j$  is included into the knapsack ( $x_j = 1$ ) or not ( $x_j = 0$ ), the MKP\_01 can be formulated as

$$\begin{aligned} & \text{Maximize} && \sum_{j=1}^n p_j x_j \\ & \text{Subject to} && \sum_{j=1}^n r_{ij} x_j \leq c_i, \quad i = 1, \dots, m, \\ & && x_j \in \{0, 1\}, \quad j = 1, \dots, n. \end{aligned} \quad (1)$$

The MKP\_01 is a well-known NP-Hard problem. There are many practical applications which can be formulated as a MKP\_01, for example, the capital budgeting problem, the cargo loading, the processor allocation in distributed systems, and the project selection. Therefore, more and more people recently focus on the research for solving the MKP\_01. In general, the solving algorithms can be divided into two kinds: exact and heuristic methods [1]. The exact methods used to employ some typical search techniques, such as Enumeration algorithm [2], Branch and Bound method [3], and Approximate Dynamic programming [4]. These methods can be only applied to some small-scaled MKP\_01 because the computation complexity is rather high. Subsequently, many heuristic search methods, including Genetic Algorithm (GA) [5, 6], Evolutionary Algorithm (EA) [7], Particle Swarm Optimization (PSO) [8], Ant Colony Optimization (ACO) [9–12], and Artificial Bee Colony (ABC) [13, 14], were proposed by simulating some natural phenomena. As these population-based methods are versatile and robust, thus they have been proved to be very effective methods. Thereinto, ABC algorithm is a recently proposed method, which employs the

mechanism of combining local and global searches to effectively solve MKPs. However, since the algorithm framework itself is flawed, there is a main bottleneck that the iteration number is too large and the convergence time is too long, which strongly restricts the development of ABC algorithm. In [15], we proposed an artificial bee colony algorithm for the MKP\_01, which introduced the pheromone into ABC algorithm and gave the corresponding transition strategy. Though there are still some problems, the preliminary experimental results in [15] are very encouraging, and these results are significant motivations for the present research. This paper conducts a further and thorough investigation along this line. In comparison with our previous work, the main new contributions of this paper include the following.

(1) Based on the researches of entomologists, a new algorithm, combining chemical communication way and behavior communication way for solving MKP\_01 (ABCPUD-MKP), has been developed. First, the paper extends our previous work, analyzes, and explicitly presents the pheromone communication mechanism. Second, the paper designs the constructing method of feasible solutions based on the new mechanism. Third, the paper introduces the special updating and diffusing strategies of the inductive pheromone. An important characteristic of the new algorithm is that the collaborations among bees can be strengthened, and the intelligent foraging behavior of honey swarm can be mimicked more faithfully by means of updating and diffusion of the inductive pheromone.

(2) Systematic experiments have been conducted to compare the proposed algorithm with the previous work and some other algorithms proposed recently in respective literatures on many instances of two benchmark data sets. Moreover, the sensitivity to algorithmic parameters and effects of different parameter selections have been experimentally investigated.

The rest of this paper is organized as follows. Section 2 provides an introduction to the artificial bee colony algorithm and the basic idea of the ABC for the MKP\_01. In Section 3, we describe our new algorithm for the MKP\_01. Computational results are presented in Section 4. Finally, we conclude the paper in Section 5.

## 2. ABC Algorithm and Its Application in the MKP\_01

*2.1. The Artificial Bee Colony (ABC) Algorithm.* The artificial bee colony algorithm is a population-based metaheuristic approach proposed recently; it finds near-optimal solutions to the difficult optimization problems by simulating the intelligent foraging behavior of a honeybee swarm. There are three groups of the foraging bees, employed ones, onlookers, and scouts. All bees that are currently exploiting food sources are called employed bees. These bees bring nectar from different food sources to their hive. Onlooker bees are those bees who are waiting in the hive for the information on food sources to be shared by the employed bees, and scout bees are those bees that are currently searching for new food sources near the hive. By dancing in a common area

of the hive, employed bees share the information on food sources with onlooker bees. The duration of a dance of an employed bee depends on the nectar content of the food source currently being exploited. After watching numerous dances, onlooker bees choose a food source according to the probability proportional to the quality of that food source. Therefore, the good food sources can attract more onlookers than the poor ones. Onlooker bees help the employed bee, who associated with the same food source, to perform the exploiting job within the neighborhood of the food source. Whenever a food source is exploited fully, it is abandoned by the employed bee associated with it. Then, the employed bee becomes a scout. The task of a scout is to look for a new food source around the hive that can be viewed as performing the job of exploration. Once a scout bee finds a new food source in its global expedition, it again becomes an employed one and continue its local exploiting job.

Based on such an intelligent foraging behavior of honey bee swarm, the Artificial Bee Colony (ABC) algorithm is proposed by Karaboga [16] and further developed in [17, 18]. The overall process of the ABC algorithm is given in Algorithm 1.

Each cycle of the search consists of three main steps. The first two steps are moving the employed and onlooker bees onto the food sources and performing local optimization in an exploiting way, and the third step employs a scout bee as an explorer to find a new food source for each food source exploited fully. A food source represents a feasible solution to the problem to be optimized. Due to the nectar amount of a food source corresponds to the quality of the solution represented by that food source, so the employed and onlooker bees make use of such a quantity to perform an exploitive search in a local area. Whenever a solution representing a food source is not improved by a predetermined number of cycles, then that food source is abandoned by its employed bee, and the employed bee becomes a scout. The number of cycles for releasing a food source is called a threshold value of *limit*. Every scout is an explorer who does not have any guidance while looking for a new food. That is, a scout may find any kind of food source. Therefore, sometimes a scout might accidentally discover more rich and entirely unknown food source. As a result of such behavior of scout bees, ABC algorithm can overcome the stagnation phenomenon of solutions which is a general problem of the stochastic search methods for solving a combinatorial optimization problem. It is important to note that exploration and exploitation processes are carried out together in ABC algorithm. More specially, employed and onlooker bees accomplish the exploitation process in the search space, while the scouts control the exploration process. In the ABC algorithm, local and global searches have been incorporated into different behaviors of different bees, and three kinds of bees work together to complete the evolution of solutions.

Since the ABC algorithm was proposed in 2005, it has been applied in many research fields in recent years, such as data clustering [19], flow shop scheduling problem [20, 21], multiobjective optimization [22], neural network training [23] and synthesis [24], image processing [25], generalized

```

1. Initialization: Initialize parameters  $\mathbf{K}$ ,  $\mathbf{N}$ ,  $\mathbf{Limit}$ ,  $C_i = 0$  ( $i = 1, \dots, \mathbf{K}$ );
2. Initial solutions: Randomly generated  $\mathbf{K}$  food sources  $\{S_i(0) \mid i = 1, \dots, \mathbf{K}\}$ ;
3. Loop:
  For  $t = 1$  to  $\mathbf{N}$  do:
  {
    (1) For  $i = 1$  to  $\mathbf{K}$  do: (employed bees select food sources and perform local searches respectively)
      {Associate each employed bee with a food source  $S_i(t)$  and compute its nectar amount;
      Find a new  $S'_i(t)$  in the neighborhood of  $S_i(t)$  and compute its nectar amount;
      Take the better one in  $\{S'_i(t), S_i(t)\}$  as a new location of the employed bee;}
    (2) For  $j = 1$  to  $\mathbf{K}$  do: (onlooker bees help employed bees to perform further local searches)
      {Select a food source  $S_j(t)$  from  $\{S_i(t)\}$  for every onlooker bee;
      Find a new  $S'_j(t)$  in the neighborhood of  $S_j(t)$  and compute its nectar amount;
      Take the better one in  $\{S'_j(t), S_j(t)\}$  as a new location of the corresponding bees;}
    (3) Exploiting new food sources (scout bees randomly carry out global searches)
      For  $i = 1$  to  $\mathbf{K}$  do: (food sources)
        {If  $S_i(t) = S_i(t - 1)$  then  $C_i = C_i + 1$ ;
        If  $C_i = \mathbf{Limit}$  then
          {Abandon the  $S_i(t)$  and the associated employed bee becomes a scout;
          Randomly generate a new  $S_i(t)$  and the scout becomes an employed bee again;
           $C_i = 0$ ;}
        (4) Memorize the best food source  $S_{\text{best}}$  found so far;
           $t = t + 1$ ;
      }
  }
4. Output: Return  $S_{\text{best}}$  while the predefined end condition is met.

```

ALGORITHM 1: ABC.

assignment problem [26], coupled ladder network [27], and nurse Rostering [28]. Studies [17, 29] have indicated that ABC algorithms have high search capability to find good solutions efficiently.

**2.2. ABC Algorithm for the MKP\_01.** Following above the ideas of ABC algorithm, Sundar et al. presented a method which integrates ABC algorithm with the MKP\_01, called ABC-MKP [13]. In the algorithm, a food source represents a feasible solution constituted a subset of objects for the MKP\_01, so the total profit value of objects in the subset is viewed as the nectar amount to evaluate the quality of food sources. Moreover, the ABC-MKP uses binary tournament selection method for selecting a food source for onlookers, that is, two different food sources are randomly selected from the food sources associated with employed bees, then the food source containing richer nectar amount among these two food sources is selected with a random probability  $b_t$  otherwise the poorer one is selected. For the ABC-MKP algorithm, determination of a new solution in the neighborhood of a solution is the most important process where two specific heuristic-based change operators and general local search are incorporated.

To determine a solution in the neighborhood of a solution  $i$ , the ABC-MKP algorithm randomly selects another solution  $s_j$  ( $s_j \neq s_i$ ), then randomly selects two distinct objects with the maximum profit values from  $s_j$  which are not present in  $s_i$  and add them to  $s_i$  which makes solution  $s_i$  infeasible. When this method fails to find even one object in  $s_j$  different from the objects of  $s_i$ , then it means that  $s_i$  and  $s_j$  are the

same solution [30]. There are two different processes for such a situation. In case of an employed bee, the employed bee abandons its associated solution to become a scout, and this scout is again made employed by looking for a new randomly generated solution. There is no further operations like change operator and local search. However, if the same solution occurs while determining a new neighborhood solution for an onlooker, then another solution  $s_j$  is selected randomly. This process is repeated until a solution  $s_j$  which is different from the solution  $s_i$  is found. And then, the process of making the infeasible solution feasible begins with a change operator. The change operator includes DROP PHASE and ADD PHASE [5]. The DROP PHASE drops objects in a way, which is combined random selection with greedy search, until the infeasible solution becomes feasible. With probability  $p_d$ , the objects of solution  $s_i$  are dropped in the increasing order of their pseudoutility ratios; otherwise, the objects of the solution  $s_i$  are dropped randomly. During ADD PHASE, objects which are not in the solution  $s_i$  are sorted in decreasing order in light of their pseudoutility ratios. Then each sorted and unselected object is checked one by one whether it can be added to the solution  $s_i$  without violating the feasibility. If so, then the object is added to the solution  $s_i$ . This process is repeated until all unselected objects are searched for inclusion.

The ABC-MKP algorithm uses the local search in 1-1 exchange and 2-1 exchange ways to improve the solution quality [31]. That is, the local search tries to repeatedly exchange one or two selected objects with an unselected object if such exchange can increase the total profit while maintaining the feasibility of the solution. In 1-1 exchange

way, the algorithm repeatedly exchanges a selected object in the order appeared in the knapsack with the unselected object of highest profit that will keep the solution feasible after the exchange. In 2-1 exchange way, the algorithm exchanges a pair of selected objects with the first unselected object which will keep the solution feasible after the exchange, and total profit will increase or remain the same. The 2-1 exchange is also repeated till a swapping move has been found or all pairs have been considered. As the local search costs much running time of ABC algorithm, therefore, the number of application of the local search is limited empirically. Moreover, 1-1 exchange and 2-1 exchange are used by a probability  $P_{ls}$  in a mutually exclusive manner, that is, a real number  $p$  is randomly generated from  $[0, 1]$ . If the  $p \leq P_{ls}$ , then 1-1 exchange with maximum five applications is used as the local search; otherwise a single 2-1 exchange is used as the local search.

### 3. ABC Algorithm Based on Pheromone Communication Mechanism for the MKP\_01

Though ABC algorithm has increased tremendously in many research topics, so far the only communication mechanism based on the dancing behaviors is employed to exchange information among bees in almost all applications of the ABC algorithm. From the view of entomologists, there are many ways to transfer information among bees, such as a behavior way (dancing), a chemical way (pheromone), and a physical way (light and sound). In this paper, we will introduce a chemical way with the inductive pheromone into the ABC algorithm for solving the MKP\_01.

**3.1. Communication Mechanism Based on the Inductive Pheromone.** In nature, an important way in which organisms can communicate with each other is through the use of pheromone. Pheromone is produced as a liquid and transmitted by direct contact as a liquid or as a vapor. The pheromone is a chemical messenger secreted by one individual which causes a specific response in other members of the same species. Ants and bees demonstrate two prominent examples of pheromone usage, which acknowledges their incredible capability to organize the behaviors of the whole colony. By means of the communication way of pheromone [32, 33], ant colony optimization (ACO) algorithm becomes one of the most well-known algorithms and has been successfully applied in several real-world problems. The success of the use of pheromone in ACO inspires us to incorporate the communication mechanism of pheromone into a new ABC algorithm, which is highly possible to improve the performance of the ABC algorithm.

In fact, research on the behavior of real bees has greatly inspired our work. Biologists have discovered that bees are also well known for communicating through the use of pheromone [34, 35]. Like ants, bees use pheromone for a number of different communication and behavior-control purposes. One pheromone may cause many different responses, depending on environmental conditions and pheromone concentration. Honeybee pheromone can

be grouped into releaser pheromone with short-term effects and primer pheromone with long-term effects; thereinto, releaser pheromone changes the behavior of the recipient. Releaser pheromones have a short-term effect, and they trigger an almost immediate behavioral response from the receiving bee. The inductive pheromone is a kind of releaser pheromone which is left by bees when they walk and is useful in searching for nectar. To mimic such behavior of real bees in some extent, we establish a new communication mechanism among bees in our new ABC algorithm, which includes some new strategies based on the inductive pheromone.

**3.2. Construction of Feasible Solutions Based on the New Mechanism.** At each iteration of our algorithm, each candidate solution associated to a scout bee is constructed by means of the inductive pheromone and heuristic information. It first randomly chooses an initial object and then iteratively adds objects chosen from an available set, which contains all the objects that can be selected without violating resource constraints.

Using the ideas of AS\_MKP [9], the quantity of pheromone laying on an object  $o_i$  is denoted by  $\tau_i(t)$  ( $t$  is the number of iterations). When  $t = 0$ ,  $\tau_i(0) = 1/\sum_{j=1}^n p_j$  ( $j = 1, \dots, n$ ). At each step of the construction of a solution, a scout  $l$  randomly selects the next object  $o_j$  within the set of *candidates* with respect to a probability  $P_j^l(t)$ . This probability is defined proportionally to a pheromone factor and a heuristic factor, that is,

$$P_j^l(t) = \begin{cases} \frac{[\tau_j(t)]^\alpha \cdot [\eta_j(S_l(t))]^\beta}{\sum_{i \in allowed_l(t)} [\tau_i(t)]^\alpha \cdot [\eta_i(S_l(t))]^\beta}, & j \in allowed_l(t) \\ 0, & \text{otherwise,} \end{cases} \quad (2)$$

where  $S_l(t)$  is a partial solution set acquired by the  $l$ -th scout at the time  $t$ ,  $allowed_l(t) \subseteq J - S_l(t)$  is the candidate set of remaining available objects,  $\eta_j(S_l(t))$  represents a local heuristic information, and the parameters  $\alpha$  and  $\beta$  determine the relative importance of pheromone trail versus heuristic factor for an object  $j$ . Thus, the higher the value of  $\tau_j(t)$  and  $\eta_j(S_l(t))$ , the more profitable it is to select object  $j$  in the partial solution.

**3.2.1. Heuristic Function.** There are many constraints in a MKP, so the heuristic factor depends on not only the set  $allowed_l(t)$  of candidate objects but also the whole set  $S_l(t)$  of selected objects. Let  $u_i(l, t) = \sum_{k \in S_l(t)} r_{ik}$  be the consumed quantity of the resource  $i$  when the bee  $l$  has selected those objects in  $S_l(t)$  at the time  $t$  and  $\gamma_i(l, t) = c_i - u_i(l, t)$  be the remaining capacity of the resource  $i$ , then the tightness of a candidate object  $j$  on the resource  $i$  can be expressed as

$$\delta_{ij}(l, t) = \frac{r_{ij}}{\gamma_i(l, t)}. \quad (3)$$

The equation represents a ratio between  $r_{ij}$  and  $\gamma_i(l, t)$ . Further, when an object is chosen to be included in  $S_l(t)$ , the average tightness on all constraints is defined as

$$\delta_j(l, t) = \frac{\sum_{i=1}^m \delta_{ij}(l, t)}{m}. \quad (4)$$

From the view of consumption of resources, the lower the tightness ratio, the more the object is profitable to be selected. Moreover, the profits  $p_j$  must be taken in account in order to get a pseudoutility measure for each candidate object. Thus, the local heuristic function for the MKP,  $\eta_j(S_l(t))$ , can be defined as follows:

$$\eta_j(S_l(t)) = \frac{p_j}{\delta_j(l, t)}. \quad (5)$$

**3.2.2. Inductive Pheromone Updating.** Once each food source (solution) has been optimized at every iteration, pheromone trails of the objects are updated. More specifically, in light of the profit value of every solution associated by employed bees, the algorithm updates the pheromone intensity for objects interrelated, the formula is as follows:

$$\tau_i(t+1) = (1 - \rho) \tau_i(t) + \Delta \tau_i(t, t+1), \quad (6)$$

$$\Delta \tau_i(t, t+1) = \sum_{l=1}^K \Delta \tau_i^l(t, t+1), \quad (7)$$

$$\Delta \tau_i^l(t, t+1) = \begin{cases} Q \cdot L(S_l), & o_i \in S_l \text{ for } l\text{th bee} \\ 0, & \text{otherwise,} \end{cases} \quad (8)$$

where  $0 < \rho \leq 1$  is a coefficient which represents pheromone evaporation, the  $\Delta \tau_i^l(t, t+1)$  represents the pheromone trail that the  $l$ th bee deposited on the object  $i$  in the iteration, and the  $\Delta \tau_i(t, t+1)$  represents the pheromone increment that all the bees deposited on the object  $i$ .  $Q$  is a constant parameter for an instance of MKP ( $Q = 1 / \sum_{j=1}^n p_j$ ), and  $L(S_l)$  is the value of the objective function of the solution  $S_l$  obtained by the  $l$ th bee.

From these descriptions about the heuristic function and pheromone updating, we can know that a value of the transition probability represents a trade-off between pseudoutility and pheromone intensity. That is, those objects which consume less resources and have more profit should be chosen with a high probability. On the other hand, if an object is included in many solutions, then it is highly desirable due to having high pheromone intensity.

**3.3. Diffusion Strategy of the Inductive Pheromone.** Using the new mechanism for solving MKPs, the key point is how to decide which components of the constructed solutions should be rewarded, and how to exploit these rewards when constructing a new solution. There are two different methods: (1) the first one is to lay pheromone trails on each object selected in  $J$ , which considers, respectively, the desirability of each object. The more frequent an object occurs in solutions, the more likely it would be selected when constructing a new

solution. (2) The other one is to lay pheromone on each pair  $(o_i, o_j)$  of different objects in  $J$ , which considers together the desirability of two objects in  $J$ . If some objects of  $J$  have been contained in a new partial solution, then the other objects that often occurred together with these objects in pairs of some solutions will be more attractive. Our algorithm combines these two ways into pheromone updating and diffusion.

**3.3.1. Associated Distance Based on Top- $k$  Strategy.** If we take a solution acquired by a bee as a transaction while solving a MKP, then we can get a transaction database in each iteration of a bee colony. Observing the transaction database, it is not difficult to find that some objects occur together with other objects in pairs. In other words, there are some frequent object pairs in the feasible solution space, which implies the relationship between two objects of each pair. Therefore, we give a definition of an associated distance between two objects.

*Definition 1* (associated distance). Let  $D$  be a database with  $K$  transactions; each transaction  $p$  ( $p = 1, 2, \dots, K$ ) represents a feasible solution  $S_p$  ( $S_p \subseteq J$ ), and  $(o_i, o_j)$  is a pair of objects in  $J$ . If the pair of  $(o_i, o_j)$  appears  $f$  ( $f = 0, 1, 2, \dots, K$ ) times in  $D$ , then we define the associated distance of this pair as:  $d_r(o_i, o_j) = 1/(f + 1)$ .

From this definition, we can observe the two properties:

- (1)  $d_r(o_i, o_j) = d_r(o_j, o_i)$ ;
- (2)  $0 < d_r(o_i, o_j) \leq 1$ .

The first property denotes the symmetry of the associated distance, which shows that the associated distance is irrelative with the order of objects. And the second property defines the value range of the associated distance; it indicates that the more frequent an object pair appears in  $D$ , the shorter the associated distance. That is, the higher the relation degree between the two objects, the smaller  $d_r(o_i, o_j)$  is. When  $d_r(o_i, o_j) = 1$ , the two objects are absolutely irrelative.

To solve an MKP with the ABC algorithm, the number of a bee colony is usually proportional to the number of objects in  $J$ . Thus, every iteration for a bee colony may produce a set of solutions (database), whose size approaches the number of objects in the MKP. Due to the frequency statistic ( $f$ ) for an object pair needs to scan the database in each iteration, so the computation costs of the associated distances for all object pairs may be very expensive for the large-scale MKP. Moreover, the most important idea in swarming intelligence is to strengthen the effect of good solutions and reduce the effect of bad solutions; thus, our ABC algorithm did not treat equivalently all solutions. Therefore, we adopt the idea of Top- $k$  query widely applied in Web search engine to emphasize the influence of good solutions. More precisely, we take the objective function of MKPs as the Top- $k$  function, select the  $k$  solutions which have the higher values of the objective function to form a transaction database  $D_{\text{Top-}k}$ , and then find the associated distances among objects from the  $D_{\text{Top-}k}$ . This method has two advantages: (1) reduce the computation complexity. When considering all solutions in each iteration,

mining the associated distances needs  $O(K \cdot C_n^2) \approx O(n^3)$ . However, If only considering Top- $k$  solutions ( $k \ll K$ ), the computation complexity is  $O(k \cdot n^2)$ . And (2) give prominence to the preference to the better solutions (i.e., Top- $k$  best solutions), thus strengthen the intellectual insight of the better solutions to the next optimization.

### 3.3.2. Pheromone Diffusion Based on Associated Distances.

In our new mechanism, the chemistry substance called pheromone is an important carrier for a bee colony to implement swarming intelligence. While the time passing, the pheromone gradually volatilizes; that is, the pheromone can diffuse around. To simulate this phenomenon, we present a pheromone diffusion model and the corresponding algorithm for solving MKPs. The basic idea is to take into account the pheromone influences among close objects when a bee constructs a feasible solution. Namely, based on coupling actions among near strength fields of pheromone diffusions, the algorithm performs decoupling compensates for the pheromone of near objects. Thus, the closer the associated distance, the stronger the coupling action is, and yet the further the associated distance, the weaker the coupling action is.

Let  $\Delta\tau_i$  be the pheromone trail on an object  $o_i$  laid by the bee colony; we can give the pheromone diffusion model based on the associated distance. Figure 1 shows the sketch map of the pheromone diffusion, where the red dot( $\bullet$ ) denotes the object  $o_i$  as an info fountain, those white dots ( $\circ$ ) denote the other different objects influenced by  $o_i$ , and different circles can be interpreted as equipotential lines of the intensity field for different locations. The figure shows that the pheromone trail diffuses around by taking an info fountain as a center of the diffusion intensity field, the closer to the info fountain the location, the stronger the field force of the intensity field is. More precisely, the pheromone influence of an info fountain on other objects will gradually reduce as the associated distance between objects becomes long. Therefore, we can give a simple diffusion model:

$$\Delta\tau_{ij} = \begin{cases} \left(\frac{1}{k+1}\right) \cdot \frac{\Delta\tau_i}{d_r(o_i, o_j)}, & \text{if } d_r(o_i, o_j) < 1, \\ 0, & \text{otherwise.} \end{cases} \quad (9)$$

Based on such a pheromone diffusion model, the behavior of selection of an object for a bee can change the pheromone intensity not only on the selected object but also on other objects related with the selected object. Thus, the pheromone updating of the object  $o_j$  in (6) can be revised as follows:

$$\tau'_i(t, t+1) = \tau_i(t, t+1) + \sum_r \Delta\tau_{ri}, \quad (10)$$

where  $r$  is one of the objects which are associated with the object  $o_i$ . This change more faithfully reflects the volatilization character of pheromone and then can lead bees to select objects in an impersonal way. Thus, the pheromone diffusion model can enhance the collaboration among bees in a colony, improve the effectiveness of the ABC algorithm, and in evidence incarnate the idea of swarm intelligence.

**3.4. Algorithm Description and Analysis.** Based on pheromone updating and diffusion, we design and achieve an ABC algorithm called ABCPUD for the MKPs. The algorithm holds the basic steps of original ABC algorithms, however adds the new strategies based on the inductive pheromone in the generation process of solutions. The overall process of ABCPUD-MKP algorithm is given in Algorithm 2.

From Algorithm 2, we can see that the ABCPUD-MKP is a hybrid algorithm which merges two communication ways among bees. Based on the behavior communication way, the employed bees and onlooker bees gradually optimize their associated solutions in solution neighbors as the iteration carrying through. On the other hand, the scout bees constructed new solutions in light of the chemical communication way which employs the pheromone accumulated by bees to induct the selection of objects. The important differences between the ABC-MKP and the ABCPUD-MKP focus on two phases, that is, exploring new food sources and generating, diffusing, and updating the pheromone only used by ABCPUD-MKP.

During exploring new food sources, the ABCPUD-MKP employs the transition probability based on pheromone trails and heuristic information to insightfully construct new solutions, which keeps high quality of the new solution. Moreover, the ABCPUD-MKP not only makes use of solutions acquired at every iteration to accumulate pheromone on objects but also emphasizes the pheromone influences among associated objects by means of the pheromone diffusion model which stresses the effect of the best Top- $k$  solutions. Therefore, the ABCPUD-MKP may lead to rapidly select the most profitable nectar source by means of the new communication mechanism.

## 4. Experimental Evaluation

To assess the performance of our algorithm (ABCPUD-MKP), we conduct a series of experiments on public data sets taken from the OR library, which are the same as those used in [5, 7, 11, 12], and so forth. The experimental platform was a PC with Pentium 4, 2.0 GHz CPU, 1G RAM and Windows XP. We test our algorithm implemented by MATLAB on some instances of 5.100 and 10.100 and compare the results with that of the ABC\_MKP and other algorithms recently proposed on the some data sets. The benchmark set 5.100 has 30 instances with  $m = 5$  constraints and  $n = 100$  objects. Similarly, the benchmark set 10.100 also has 30 instances with  $m = 10$  constraints and  $n = 100$  objects. All algorithms are executed 10 times on each instance with a different random seed.

By large numbers of experiments, the main parameters were set as follows:  $\alpha = 1$ ,  $\beta = 5$ ,  $N = 10000$ ,  $p_d = 0.3$ ,  $p_{IS} = 0.99$ ,  $b_t = 0.9$ ,  $K = 80$ ,  $\rho = 0.3$ ,  $k = 25$ , and  $limit = 80$ . Here  $K$ ,  $k$ ,  $\rho$ , and  $limit$  are parameters which mainly affect performances of ABCPUD-MKP. How to select their values will be emphatically discussed in the following subsection.

**4.1. Performance Analysis of ABCPUD-MKP.** We study the factors that affect the performance of ABCPUD-MKP. Particularly, we wish to investigate the contributions of the

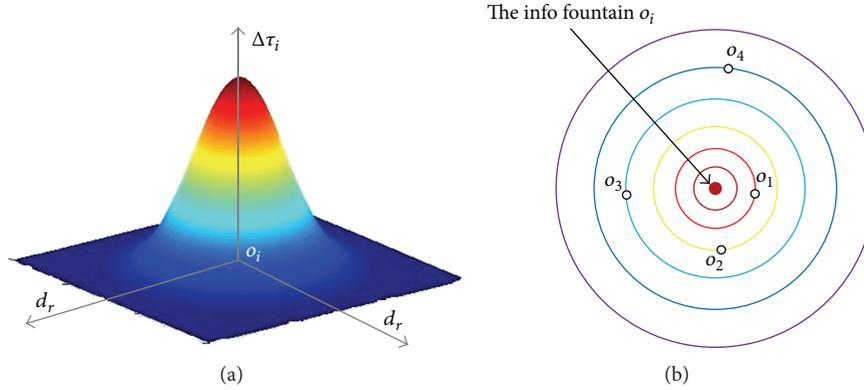


FIGURE 1: The sketch map of the pheromone diffusion of  $O_i$  and its intensity field. (a) The relationship between the intensity  $\Delta\tau_i$  and the associated distance  $d_r$  for the info fountain  $O_i$ . (b) The corresponding intensity field and areas of the pheromone diffusion.

pheromone mechanism and the effects of different parameter selection.

(1) *Contributions of the Pheromone Mechanism.* Based on the same running experiment and implemented tools, we developed three algorithms to solve the MKP\_01 on some different instances. The three algorithms were called the original ABC-MKP, the ABCPU-MKP (with the pheromone updating), and the ABCPUD-MKP (with the pheromone updating and diffusing), respectively. Table 1 gives the experimental results of three algorithms after running same computational time, where we evaluate the algorithm performance by means of three measures such as *Best*, *Avg.*, and *Num*. *Best* denotes the profit value of the best solution obtained by the corresponding algorithm over 10 executions. *Avg.* indicates the mean  $\mu$  and the standard deviation  $\sigma$  of the profit values over 10 executions independently carried out by the corresponding algorithm. *Num.* is the smallest number of the iterations when the best solution was found in 10 trails.

From the table, it is seen that these three algorithms are able to obtain the best solutions known on 10 instances. As for the *Avg.*, ABCPU-MKP produces better results than that of ABC-MKP except for the instance of 5.100.03, and ABCPUD-MKP produces the best results among the three algorithms. From the iteration process, both ABCPU-MKP and ABCPUD-MKP show better performance than ABC-MKP; the latter is more outstanding. Therefore, we can draw the conclusion that the new pheromone mechanism introduced in the paper can evidently improve the ABC-MKP algorithm on the convergence performance while keeping good solution quality. More specifically, (1) the solution construction based on the inductive pheromone not only effectively enhances the convergence performance but also improves the solution quality on a majority of instances tested, which shows that the inductive pheromone is an important media to improve the bee communication in finding better solutions. (2) The new diffusion mechanism can also improve the convergence performance while keeping the solution quality. This suggests that the strategy based on the diffusion model can effectively strengthen the process of solution construction and save iteration times. It is obvious

that both of the aforementioned strategies are effective in the improvement of the performance of the ABC-MKP algorithm. This fact encourages us to put both strategies into our algorithm (ABCPUD-MKP) to get even better results.

(2) *Effects of Different Parameter Selection.* In this experiment, we give a solving example to show how to determine the parameter value of ABCPUD-MKP algorithm. The experiment process is that we run ABCPUD-MKP with different parameter settings on the same instance (5.100.06) and acquire the best parameter value by comparing experimental results. During this experimentation, the value of a single parameter is changed while keeping the values of other parameters fixed.

ABC algorithm is a swarm optimization algorithm; the population size of a bee colony determines the number of solutions at each iteration. Figure 2 summarizes the performance of ABCPUD-MKP with 10 different bee colony sizes ( $K$ ). The best value, the worst value, and the average value are, respectively, corresponding the highest, lowest, and mean profit value of solutions obtained in the 10 trails (shown in Figure 2(a)). Figure 2(b) illustrates the results about the running time. The median of each bar is the mean, and the height of the bar is the standard deviation, which are obtained from 10 trails.

A large bee colony means that more initial search points are employed, and then, as reflected by best, worst, and average values in Figure 4, better solutions are obtained than in a smaller bee colony. However, after a sufficient value for the colony size, any increment does not improve the solution performance of algorithms. On the contrary, the search time in each iteration will increase as the size of the bee colony increases. To acquire a balance between getting a better solution and using less time, we recommend a bee colony size of 80 ( $K = 80$ ).

As described in Section 3.2, the coefficient  $\rho$  is a value smaller than 1 to avoid unlimited accumulation of pheromone. In the constructing solution process, the value of  $\rho$  controls the balance between exploration and exploitation processes. Therefore, it is an important parameter for the new mechanism based on the inductive pheromone. To investigate

**1. Initialization:** Initialize parameters  $K, N, Limit, \alpha, \beta, Q, \tau_0, C_i = 0$  ( $i = 1, \dots, K$ ) and so on;

**2. Initial solutions:** Randomly generated  $K$  food sources  $\{S_i(0) \mid i = 1, \dots, K\}$ ;

**3. Loop:**

For  $t = 1$  to  $N$  do:

{

(1) For  $i = 1$  to  $K$  do: (local searches performed by employed bees)

{Associate each employed bee with a food source  $S_i(t)$  and compute its nectar amount;

Find a new  $S'_i(t)$  in the neighborhood of  $S_i(t)$  and compute its nectar amount;

Take the better one in  $\{S'_i(t), S_i(t)\}$  as a new location of the employed bee;}

(2) For  $j = 1$  to  $K$  do: (further local searches performed by onlooker bees)

{Select a food source  $S_j(t)$  from  $\{S_i(t)\}$  for every onlooker bee;

Find a new  $S'_j(t)$  in the neighborhood of  $S_j(t)$  and compute its nectar amount;

Take the better one in  $\{S'_j(t), S_j(t)\}$  as a new location of the corresponding bees;}

**(3) Exploiting new food sources** (global searches with a guidance performed by scout bees)

For  $l = 1$  to  $K$  do: (food sources)

{If  $S_l(t) = S_l(t-1)$  then  $C_l = C_l + 1$ ;

If  $C_l = Limit$  then

{Abandon food source  $l$  and the associated employed bee becomes a scout;

$S_l(t) = \{\}, allowed_l = J$ ;

**Repeat** (constructing a new solution)

Select an object  $j$  with  $P'_j(t)$  given by (2);

Add the object into the current solution:  $S_l(t) = S_l(t) + \{o_j\}$ ;

$allowed_l = allowed_l - \{o_j\}$ ;

**Until**  $allowed_l$  is empty

The scout bee becomes again an employed bee;

$C_l = 0$ ;}}

**(4) Generating, diffusing and updating the pheromone**

For each object  $o_i$  in  $J$

Calculate the  $\Delta\tau_i$  according to (7) and (8).

Select the Top- $k$  solutions from this iteration, and obtain  $D_{Top-k}$ ;

For each solution  $S_w \in D_{Top-k}$

{For each pair of objects  $c_{ij}$  in  $S_w$

$c_{ij}.count++$ ;

Calculate the associated distance  $d_r(o_i, o_j)$

Calculate  $\Delta\tau_{ij}$  according to (9);}

For each object  $o_i$  in  $J$

Update the trail level  $\tau_j$  on all objects according to (6) and (10);

(5) Perform the local optimization in 1-1 or 2-1 exchange ways;

(6) Memorize the best food source  $S_{best}$  found so far;

$t = t + 1$ ;

}

**4. Output:** Return  $S_{best}$  while the predefined end condition is met.

ALGORITHM 2: ABCPUD-MKP.

TABLE 1: The contributions of the pheromone mechanism on ABC-MKP.

Instances	Best known	ABC-MKP			ABCPUD-MKP			ABCPUD-MKP		
		Best	Avg.	Num.	Best	Avg.	Num.	Best	Avg.	Num.
5.100.00	24381	24381	24381.0 ± 0.0	238	24381	24381.0 ± 0.0	220	24381	24381.0 ± 0.0	<b>178</b>
5.100.01	24274	24274	24274.0 ± 0.0	36	24274	24274.0 ± 0.0	32	24274	24274.0 ± 0.0	<b>20</b>
5.100.02	23551	23551	23539.3 ± 4.1	4021	23551	23548.0 ± 5.5	2251	23551	23550.0 ± 3.9	<b>354</b>
5.100.03	23534	23534	23534.0 ± 0.0	2356	23534	23527.0 ± 3.5	2202	23534	23534.0 ± 0.0	<b>1538</b>
5.100.04	23991	23991	23973.2 ± 14.8	1746	23991	23982.4 ± 13.7	1689	23991	23986.0 ± 10.5	<b>1622</b>
5.100.05	24613	24613	24613.0 ± 0.0	110	24613	24613.0 ± 0.0	76	24613	24613.0 ± 0.0	<b>46</b>
5.100.06	25591	25591	25591.0 ± 0.0	252	25591	25591.0 ± 0.0	224	25591	25591.0 ± 0.0	<b>65</b>
5.100.07	23410	23410	23410.0 ± 0.0	409	23410	23410.0 ± 0.0	304	23410	23410.0 ± 0.0	<b>259</b>
5.100.08	24216	24216	24216.0 ± 0.0	1051	24216	24216.0 ± 0.0	799	24216	24216.0 ± 0.0	<b>755</b>
5.100.09	24411	24411	24411.0 ± 0.0	189	24411	24411.0 ± 0.0	97	24411	24411.0 ± 0.0	<b>81</b>

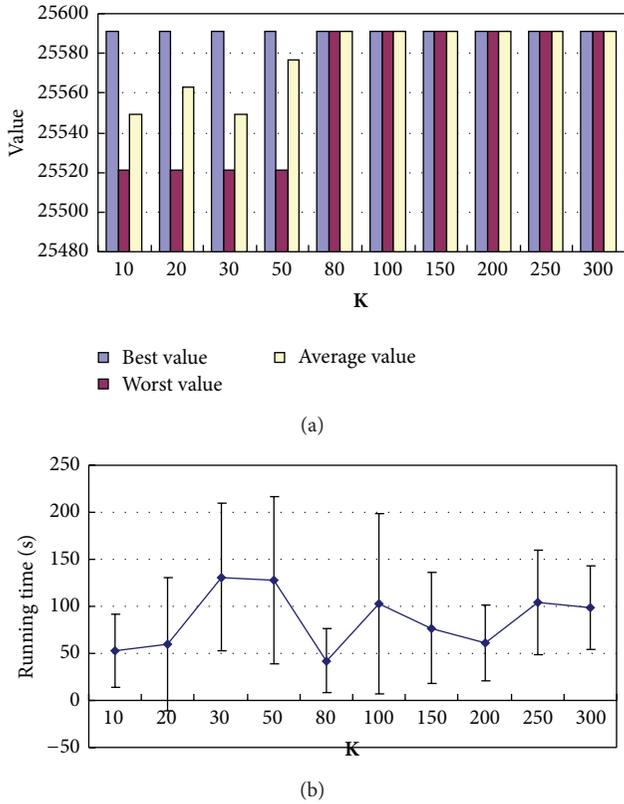


FIGURE 2: Comparisons of the results for different bee colony sizes ( $K$ ). (a) Best, worst, and average values for different  $K$ . (b) The runtime for different  $K$ .

the effect of  $\rho$  on ABCPUD-MKP, we perform experiments using different values of  $\rho$ . The results are presented in Figure 3.

From Figure 3(a), we notice that the optimum values on best, worse, and average values can be obtained for different parameter values of  $\rho$ . That is, the solution of ABCPUD-MKP is insensitive to the parameter  $\rho$ . However, the significant difference focuses on the running time shown in Figure 3(b). Because the smallest running time is achieved for  $\rho = 0.3$  among all testings, we select a  $\rho$  value of 0.3 for our ABCPUD-MKP.

In our algorithm, we introduce a pheromone diffusion model based on associated distances. To save computation costs and strengthen the effect of good solutions, we adopt the Top- $k$  strategy to compute the associated distances of every pair of objects. In this section, we perform experiments using different values of  $k$  to investigate the effect of the Top- $k$  strategy on ABCPUD-MKP. Figure 6 shows the experimental results. We observe that there is no difference among the three profit values in Figure 4(a), and only there are some differences in running time for different values of  $k$  in Figure 4(b). To quickly get the best result, we can select  $k = 25$ .

If a solution does not improve for a predetermined number of iterations ( $limit$ ), then the solution will be abandoned by the employed bee associated, and a new solution

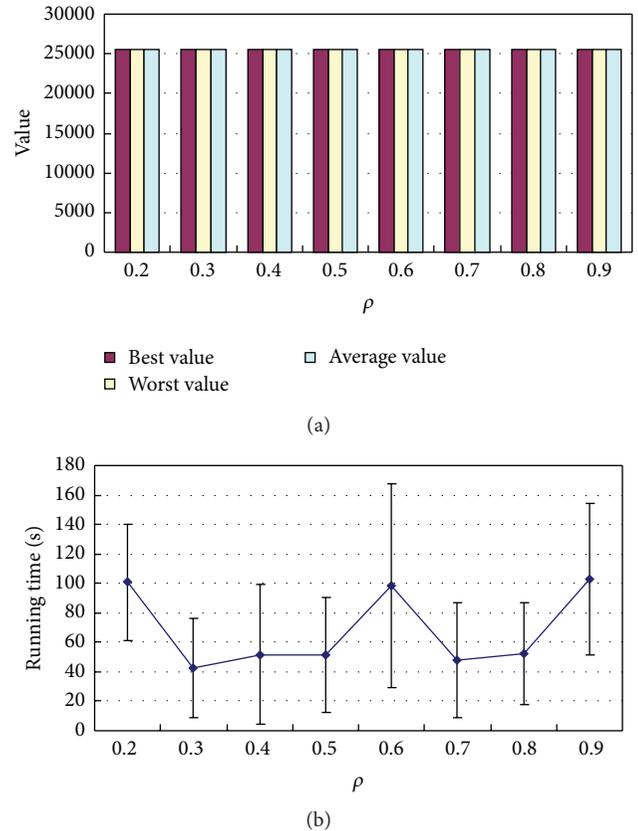
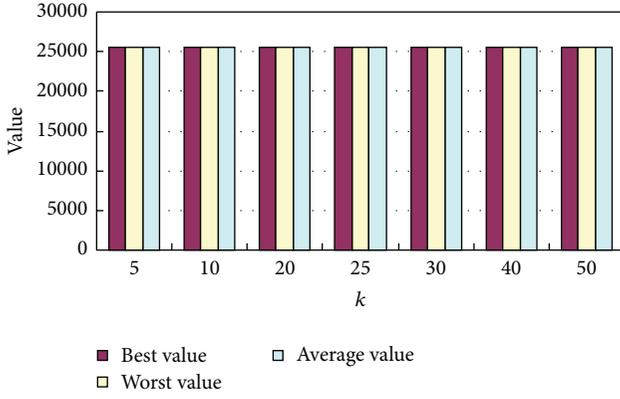


FIGURE 3: Comparisons of the results for different pheromone parameters ( $\rho$ ). (a) Best, worst, and average values for different  $\rho$ . (b) The runtime for different  $\rho$ .

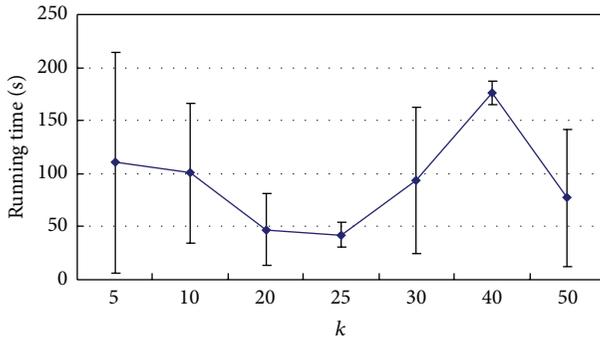
will be generated. This is a key strategy to deal with the solution stagnation in ABC algorithm. To study the effect of different abandoned frequencies on ABCPUD-MKP, we perform experiments using different values of  $limit$ . The experimental results are presented in Figure 5.

When  $limit$  is too small or too large, the results obtained by our algorithm are worse than those produced by using the moderate values of  $limit$ . This shows that an appropriate frequency of new solution production has useful effect on the running time and the profit values, which can perform some explorations to improve the search ability of the algorithm. However, the balance between exploration and exploitation processes will be broken whether  $limit$  is too small or too large, which will produce worse solutions and cost much more running time. Thus, we recommend  $limit = 80$  after considering effects on factors of the running time and solution quality.

**4.2. Comparing ABCPUD-MKP with ABC-MKP.** We compare the ABCPUD-MKP with ABC-MKP on many instances. As space is limited, Table 2 only provides a summary of the performance comparison on 6 different instances. Both algorithms are independently executed 10 times for each instance, and the figures are, therefore, the average values of 10 trails.



(a)

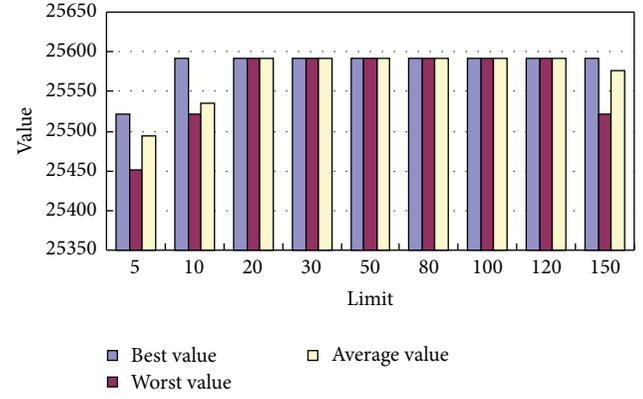


(b)

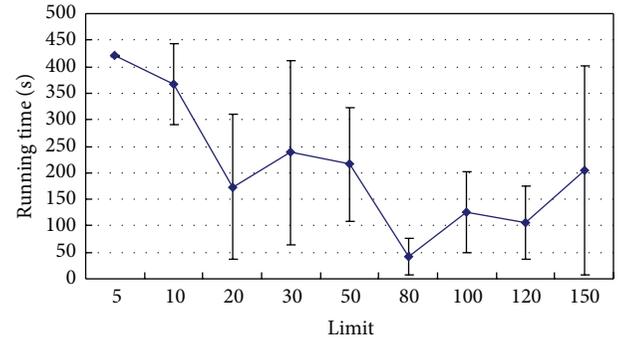
FIGURE 4: Comparisons of the results for different Top- $k$  parameters ( $k$ ). (a) Best, worst, and average values for different  $k$ . (b) The runtime for different  $k$ .

In Table 2, the meanings of *Best*, *Avg.*, and *Num.* are same as those of Table 1 while the meaning of *Time* is the same as that of Figure 2. *Hit.* represents the number of the best solution obtained in 10 trails. Moreover, numbers in parentheses of the *best* column are the best results known for each corresponding instance, and numbers in parentheses of the *Num.* and *Time* columns are the smallest numbers of the iterations and the shortest running time when the best result was obtained. Compared with ABC-MKP, ABCPUD-MKP can always find better or equally good solutions for all the instances in terms of both profit values. Since the *Num.* is reduced on all instances, hence the *Time* of ABCPUD-MKP is greatly improved. Moreover, the ABCPUD-MKP algorithm also outperforms ABC-MKP algorithm according to the item of *Hit*.

In Figure 6, we compare the iteration numbers and the runtime of an iteration of two algorithms on ten instances of 10.100. For each algorithm, we record the iteration number averaged over 10 runs and compute the runtime of an iteration averaged over 10 runs when obtaining the profit values shown as Table 1 on respective instances. From Figure 6(a), we observe that the iteration numbers of both algorithms vary with different instances. Furthermore, the new pheromone mechanism adopted by ABCPUD-MKP can improve the iteration process for all instances. However, the new pheromone



(a)



(b)

FIGURE 5: Comparisons of the results for different abandoned frequencies (*limit*). (a) Best, worst, and average values for different *limit*. (b) The runtime for different *limit*.

mechanism will increase some computation cost; thus, the runtime per an iteration of ABCPUD-MKP may be longer than that of ABC-MKP on some instances as shown in Figure 6(b).

Figure 7 gives the time performance comparison between two algorithms, which corresponds to Figure 6. We can see that ABCPUD-MKP performs better than ABC-MKP in terms of the runtime on all instances. The main reason is that ABCPUD-MKP can effectively decrease the iteration number. As shown in Figure 7, the advantage in runtime is the most remarkable on almost all instances. This denotes that the pheromone updating and diffusing can accelerate the optimization process by strengthening the communication among bees.

**4.3. Comparing ABCPUD-MKP with Other Algorithms.** To further evaluate the new algorithm, we compare the solution performance of different algorithms on some large problems. Our main objective in this section is to determine whether ABCPUD-MKP is more efficient and effective on comparable performances than these state-of-the-art approaches which employed various stochastic search schemes in recent years.

We compare the performance of ABCPUD-MKP with that of GA-MKP [5], B&B-EA-MKP [7], Ant-knapsack [11],

TABLE 2: The results for two algorithms on some instances.

Instance statistic	Algorithm	
	ABC-MKP	ABCPUD-MKP
Best	<b>24274 (24274)</b>	<b>24274 (24274)</b>
Avg.	<b>24274 ± 0.0</b>	<b>24274 ± 0.0</b>
5.100.01 Num.	130.25 ± 108.78 (53)	<b>87.60 ± 74.02 (35)</b>
Hit.	<b>10</b>	<b>10</b>
Time (s)	12.50 ± 10.87 (4.88)	<b>7.44 ± 6.56 (2.70)</b>
Best	<b>23534 (23534)</b>	<b>23534 (23534)</b>
Avg.	<b>23534 ± 0.0</b>	<b>23534 ± 0.0</b>
5.100.03 Num.	4118.33 ± 782.49 (2356)	<b>2571.0 ± 551.54 (2181)</b>
Hit.	<b>10</b>	<b>10</b>
Time (s)	389.45 ± 70.26 (223.03)	<b>216.21 ± 51.11 (197.92)</b>
Best	<b>24613 (24613)</b>	<b>24613 (24613)</b>
Avg.	<b>24613 ± 0.0</b>	<b>24613 ± 0.0</b>
5.100.05 Num.	178.8 ± 63.33 (110)	<b>96.0 ± 21.74 (74)</b>
Hit.	<b>10</b>	<b>10</b>
Time (s)	14.24 ± 5.05 (8.67)	<b>8.03 ± 1.80 (5.89)</b>
Best	<b>22801 (22801)</b>	<b>22801 (22801)</b>
Avg.	<b>22801 ± 0.0</b>	<b>22801 ± 0.0</b>
10.100.01 Num.	3824.67 ± 2959.10 (844)	<b>3161.79 ± 2331.48 (103)</b>
Hit.	<b>10</b>	<b>10</b>
Time (s)	318.72 ± 304.74 (88.52)	<b>276.22 ± 244.62 (10.41)</b>
Best	<b>22772 (22772)</b>	<b>22772 (22772)</b>
Avg.	<b>22772 ± 0.0</b>	<b>22772 ± 0.0</b>
10.100.03 Num.	5707.0 ± 1035.23 (4483)	<b>4441.5 ± 1469.34 (2855)</b>
Hit.	<b>10</b>	<b>10</b>
Time (s)	506.3 ± 97.47 (400.33)	<b>399.15 ± 133.47 (254.10)</b>
Best	<b>22777 (22777)</b>	<b>22777 (22777)</b>
Avg.	22761.8 ± 29.50	<b>22771.3 ± 19.30</b>
10.100.05 Num.	5887.6 ± 2743.9 (585)	<b>3706.8 ± 2345.87 (258)</b>
Hit.	7	<b>9</b>
Time (s)	633.32 ± 349.37 (65.16)	<b>338.09 ± 209.85 (23.07)</b>

ACOMPD-MKP [12], and ABC-MKP. GA-MKP incorporates a heuristic operator which utilizes problem-specific knowledge into the genetic algorithm to solve the multidimensional knapsack problem. B&B-EA-MKP is a hybrid approach which cooperates an evolutionary algorithm (EA) with the branch and bound method (B&B) by exchanging information. Ant-knapsack is an ACO algorithm for the multidimensional knapsack problem, which lays pheromone trails not only on the edges of the visited paths but on all edges connecting any pair of nodes belonging to the solution. ACOMPD-MKP proposed in our prophase research is a novel Ant algorithm, which employs a pheromone diffusion model and a solution mutation strategy to get high quality results. Moreover, as described in Section 2.2, ABC-MKP is a new algorithm which applies ABC algorithm to solve the multidimensional knapsack problem. Therefore, these algorithms are stochastic search approaches, which need to be executed time after time for each testing instance.

In Table 3, we summarize the comparative results, where the results of GA-MKP, B&B-EA-MKP, Ant-knapsack, and

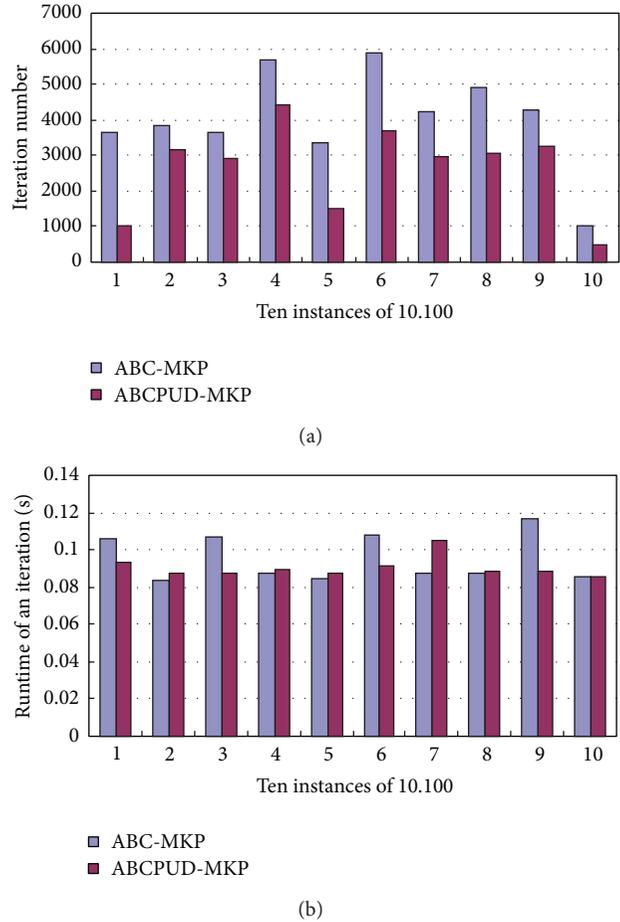


FIGURE 6: Comparison of the iteration performance on different instances: (a) the iteration numbers, (b) the runtime of an iteration.

ACOMPD-MKP are derived from the respective literature (represents no testing) while the results of ABC-MKP and ABCPUD-MKP algorithms obtained by our experiment over 10 runs.

As it can be seen, ABCPUD-MKP always finds all best solutions on six instances in terms of the profit value while other algorithms only find some best solutions. For instance, ACOMPD-MKP can find the best solutions on five instances except for 5.500.29, B&B-EA-MKP can find best solutions on 5.100.00, 10.100.00, 5.250.00, and 5.500.29 instances. As for the average solutions, ABCPUD-MKP is also competitive with B&B-EA-MKP which provides the highest quality results so far. More specifically, B&B-EA-MKP produces better results than that of ABCPUD-MKP on 5.250.00, 5.250.29, and 5.500.29 instances, and ABCPUD-MKP produces better results than that of B&B-EA-MKP on 5.100.29 and 10.100.00 instances while both algorithms can obtain the best results on 5.100.00 instance. Moreover, the results of average solutions which ABCPUD-MKP produces on 5.250.00, 5.250.29, and 5.500.29 instances are only inferior to that of B&B-EA-MKP. Thus, ABCPUD-MKP is the most outstanding algorithm in terms of the solution quality.

TABLE 3: Solution comparison among GA [5], B&B-GA [7], Ant-knapsack [11], ACOMPD-MKP [12], ABC-MKP, and ABCPUD-MKP algorithms.

Algorithm	Solution	Instance					
		5.100.00	5.100.29	10.100.00	5.250.00	5.250.29	5.500.29
GA-MKP	Best	<b>24381</b>	59960	<b>23064</b>	59243	154668	299885
	Avg.	<b>24381 ± 0.0</b>	59960 ± 0.0	23050.2 ± 19.2	59211.7 ± 18.0	154626.2 ± 31.7	299842.7 ± 26.9
B&B-EA-MKP	Best	<b>24381</b>	59965	<b>23064</b>	<b>59312</b>	154668	<b>299904</b>
	Avg.	<b>24381 ± 0.0</b>	59965 ± 0.0	23059.1 ± 3.2	<b>59305.1 ± 20.7</b>	<b>154668 ± 0</b>	<b>299902.3 ± 5.1</b>
Ant-knapsack	Best	<b>24381</b>	59965	<b>23064</b>	—	—	—
	Avg.	24342 ± 29.3	59958 ± 8.4	23016 ± 42.2	—	—	—
ACOMPd-MKP	Best	<b>24381</b>	<b>60117</b>	<b>23064</b>	<b>59312</b>	<b>154735</b>	299853
	Avg.	24362.8 ± 25.5	59979.1 ± 34.4	23051.2 ± 16.7	59152 ± 67.3	154622.3 ± 48.3	299817.0 ± 24.5
ABC-MKP	Best	<b>24381</b>	59965	<b>23064</b>	59243	154668	299885
	Avg.	<b>24381 ± 0.0</b>	59961.0 ± 2.1	23059.1 ± 3.9	59215.2 ± 17.3	154637.4 ± 15.5	299804.4 ± 59.1
ABCPUD-MKP	Best	<b>24381</b>	<b>60117</b>	<b>23064</b>	<b>59312</b>	<b>154735</b>	<b>299904</b>
	Avg.	<b>24381 ± 0.0</b>	<b>60015.4 ± 62.6</b>	<b>23060.4 ± 4.1</b>	59222.6 ± 35.5	154649.2 ± 34.2	299848.0 ± 42.4

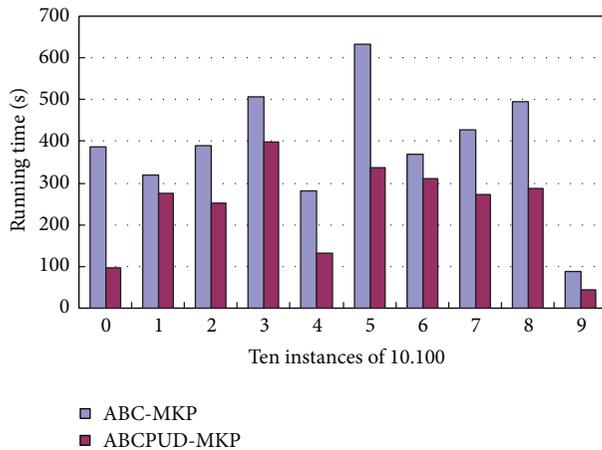


FIGURE 7: Comparison of the time performance on different instances.

## 5. Conclusion

In this paper, we propose an ABC algorithm, ABCPUD-MKP, to solve the 0-1 multidimensional knapsack problem effectively and efficiently. This algorithm differs from general ABC algorithms in the fact that a new communication way based on an inductive pheromone is introduced. More specifically, the new algorithm applies the pheromone updating and diffusion to extend the ABC algorithm. The new algorithm has been tested on many instances of the MKP problem with encouraging results: the new algorithm is superior in terms of the computational time on all instances compared to ABC-MKP algorithm, while it can also achieve best solution quality on all instances. We have also presented the performance comparison among GA-MKP, B&B-EA-MKP, Ant-knapsack, ACOMPd-MKP, ABC-MKP, and ABCPUD-MKP on some large problems and found that ABCPUD-MKP can find the best optimization solutions on all problems tested.

The new mechanism employs the chemical communication way to strength the collaboration among bees, which not

only can keep the balance between exploitation and exploration, but also can effectively look into promising regions of the search space. Thus, the new mechanism is equally significant for ABC algorithms to tackle difficult combinatorial problems. Our future work is to extend our study to other NP-hard problems such as traveling salesman problem and Bayesian network structure learning.

## Acknowledgments

This work is partly supported by the NSFC research program (61033004), 973 program (2011CB302703), and the Beijing Natural Science Foundation (4102010).

## References

- [1] A. Fréville, "The multidimensional 0-1 knapsack problem: an overview," *European Journal of Operational Research*, vol. 155, no. 1, pp. 1–21, 2004.
- [2] A. V. Cabot, "An enumeration algorithm for knapsack problems," *Operations Research*, vol. 18, pp. 306–311, 1970.
- [3] W. Shih, "A branch and bound method for the multiconstraint zero-one knapsack problem," *Journal of the Operations Research Society*, vol. 30, pp. 369–378, 1979.
- [4] D. Bertsimas and R. Demir, "An approximate dynamic programming approach to multidimensional knapsack problems," *Management Science*, vol. 48, no. 4, pp. 550–565, 2002.
- [5] P. C. Chu and J. E. Beasley, "A genetic algorithm for the multidimensional knapsack problem," *Journal of Heuristics*, vol. 4, no. 1, pp. 63–86, 1998.
- [6] J.-C. Bai, H.-Y. Chang, and Y. Yi, "An partheno-genetic algorithm for multidimensional knapsack problem," in *Proceedings of the International Conference on Machine Learning and Cybernetics (ICMLC '05)*, pp. 2962–2964, August 2005.
- [7] J. E. Gallardo, C. Cotta, and A. J. Fernández, "Solving the multidimensional knapsack problem using an evolutionary algorithm hybridized with branch and bound," in *Proceedings of the 1st International Work-Conference on the Interplay Between Natural and Artificial Computation (IWINAC '05)*, pp. 21–30, June 2005.

- [8] M. Kong and P. Tian, "Application of the particle swarm optimization to the multidimensional knapsack problem, Artificial Intelligence and Soft Computing," in *Proceedings of the 8th International Conference (ICAISC '06)*, pp. 1140–1149, 2006.
- [9] M. Kong, P. Tian, and Y. Kao, "A new ant colony optimization algorithm for the multidimensional Knapsack problem," *Computers and Operations Research*, vol. 35, no. 8, pp. 2672–2683, 2008.
- [10] P. H. Rafael and D. Nikitas, "On the Performance of the Ant Colony System for Solving the Multidimensional Knapsack Problem," in *Proceedings of the IEEE Pacific Rim Conference on Communications Computers and Signal Processing (PACRIM '03)*, pp. 338–341, August 2003.
- [11] I. Alaya, C. Solnon, and K. Ghedira, "Ant algorithm for the multidimensional knapsack problem," in *Proceedings of International Conference on Bioinspired Methods and their Applications*, pp. 63–72, 2004.
- [12] J. Z. Ji, Z. Huang, and C. N. Liu, "An ant colony optimization algorithm based on mutation and pheromone diffusion for the multidimensional knapsack problems," *Jisuanji Yanjiu yu Fazhan/Computer Research and Development*, vol. 46, no. 4, pp. 644–654, 2009.
- [13] S. Sundar, A. Singh, and A. Rossi, "An artificial bee colony algorithm for the 0-1 multidimensional knapsack problem," *Communications in Computer and Information Science*, vol. 94, no. 1, pp. 141–151, 2010.
- [14] S. Pulikanti and A. Singh, "An artificial bee colony algorithm for the quadratic knapsack problem," in *Proceedings of the International Conference on Neural Information Processing (ICONIP '09)*, vol. 5864 of *Lecture Notes in Computer Science*, pp. 196–205, 2009.
- [15] H. K. Wei, J. Z. Ji, Y. F. Qin, Y. M. Wang, and C. N. Liu, "A novel artificial bee colony algorithm based on attraction pheromone for the multidimensional knapsack problems, artificial," in *Proceedings of the Artificial Intelligence and Computational Intelligence (AICI '11) (part II)*, vol. 7003 of *Lecture Notes in Computer Science*, pp. 1–10, 2011.
- [16] D. Karaboga, "An idea based on honey bee swarm for numerical optimization," Tech. Rep. TR06, Computer Engineering Department, Erciyes University, Kayseri, Turkey, 2005.
- [17] D. Karaboga and B. Basturk, "On the performance of artificial bee colony (ABC) algorithm," *Applied Soft Computing Journal*, vol. 8, no. 1, pp. 687–697, 2008.
- [18] W.-F. Gao and S.-Y. Liu, "A modified artificial bee colony algorithm," *Computers and Operations Research*, vol. 39, no. 3, pp. 687–697, 2012.
- [19] X. H. Yan, Y. L. Zhu, W. P. Zou, and L. Wang, "A new approach for data clustering using hybrid artificial bee colony algorithm," *Neurocomputing*, vol. 97, pp. 241–250, 2012.
- [20] Q.-K. Pan, M. Fatih Tasgetiren, P. N. Suganthan, and T. J. Chua, "A discrete artificial bee colony algorithm for the lot-streaming flow shop scheduling problem," *Information Sciences*, vol. 181, no. 12, pp. 2455–2468, 2011.
- [21] Q. K. Pan, L. Wang, K. Mao, J. H. Zhao, and M. Zhang, "An effective artificial bee colony algorithm for a real-world hybrid flowshop problem in steelmaking process," *IEEE Transactions on Automation Science and Engineering*, vol. 10, no. 2, pp. 307–322, 2013.
- [22] S. N. Omkar, J. Senthilnath, R. Khandelwal, G. Narayana Naik, and S. Gopalakrishnan, "Artificial Bee Colony (ABC) for multi-objective design optimization of composite structures," *Applied Soft Computing Journal*, vol. 11, no. 1, pp. 489–499, 2011.
- [23] D. Karaboga, B. Basturk, and C. Ozturk, "Artificial bee colony (ABC) optimization algorithm for training feed-forward neural networks," *Modeling Decisions for Artificial Intelligence*, vol. 4617, pp. 318–329, 2007.
- [24] P. Y. Kumbhar and P. S. Krishnan, "Use of Artificial Bee Colony (ABC) algorithm in artificial neural network synthesis," *International Journal of Advanced Engineering Sciences and Technologies*, vol. 11, no. 1, pp. 162–171, 2011.
- [25] C. Xu and H. Duan, "Artificial bee colony (ABC) optimized edge potential function (EPF) approach to target recognition for low-altitude aircraft," *Pattern Recognition Letters*, vol. 31, no. 13, pp. 1759–1772, 2010.
- [26] B. Adil, Ö. Lale, and T. Pinar, *Artificial bee colony algorithm and its application to generalized assignment problem, swarm intelligence: focus on ant and particle swarm optimization*, Book edited by: F. T. S. Chan and M. K. Tiwari, Itech Education and Publishing, Vienna, Austria, 2007.
- [27] P. Mukherjee and L. Satish, "Construction of equivalent circuit of a single and isolated transformer winding from FRA data using the ABC algorithm," *IEEE Transactions on Power Delivery*, vol. 27, no. 2, pp. 963–970, 2012.
- [28] N. Todorovic and S. Petrovic, "Bee colony optimization algorithm for nurse rostering," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 43, no. 2, pp. 467–473, 2013.
- [29] D. Karaboga and B. Akay, "A comparative study of Artificial Bee Colony algorithm," *Applied Mathematics and Computation*, vol. 214, no. 1, pp. 108–132, 2009.
- [30] A. Singh, "An artificial bee colony algorithm for the leaf-constrained minimum spanning tree problem," *Applied Soft Computing Journal*, vol. 9, no. 2, pp. 625–631, 2009.
- [31] A. Singh and A. K. Gupta, "Two heuristics for the one-dimensional bin-packing problem," *OR Spectrum*, vol. 29, no. 4, pp. 765–781, 2007.
- [32] M. Dorigo, V. Maniezzo, and A. Colnari, "Ant system: optimization by a colony of cooperating agents," *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 26, no. 1, pp. 29–41, 1996.
- [33] C. Blum and M. Dorigo, "The Hyper-Cube Framework for Ant Colony Optimization," *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 34, no. 2, pp. 1161–1172, 2004.
- [34] M. F. Ali and E. D. Morgan, "Chemical communication in insect communities: a guide to insect pheromones with special emphasis on social insects," *Biological Reviews of the Cambridge Philosophical Society*, vol. 65, no. 3, pp. 227–247, 1990.
- [35] R. K. VanderMeer, M. D. Breed, K. E. Espelie, and M. L. Winston, *Pheromone Communication in Social Insects*, Westview Press, 1998.

## Research Article

# Visual Object Tracking Based on 2DPCA and ML

Ming-Xin Jiang,<sup>1,2</sup> Min Li,<sup>1</sup> and Hong-Yu Wang<sup>2</sup>

<sup>1</sup> School of Information & Communication Engineering, Dalian Nationalities University, Dalian 116600, China

<sup>2</sup> School of Information & Communication Engineering, Dalian University of Technology, Dalian 116600, China

Correspondence should be addressed to Min Li; [limin@dlnu.edu.cn](mailto:limin@dlnu.edu.cn)

Received 7 March 2013; Accepted 23 May 2013

Academic Editor: Yudong Zhang

Copyright © 2013 Ming-Xin Jiang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We present a novel visual object tracking algorithm based on two-dimensional principal component analysis (2DPCA) and maximum likelihood estimation (MLE). Firstly, we introduce regularization into the 2DPCA reconstruction and develop an iterative algorithm to represent an object by 2DPCA bases. Secondly, the model of sparsity constrained MLE is established. Abnormal pixels in the samples will be assigned with low weights to reduce their effects on the tracking algorithm. The object tracking results are obtained by using Bayesian maximum a posteriori (MAP) probability estimation. Finally, to further reduce tracking drift, we employ a template update strategy which combines incremental subspace learning and the error matrix. This strategy adapts the template to the appearance change of the target and reduces the influence of the occluded target template as well. Compared with other popular methods, our method reduces the computational complexity and is very robust to abnormal changes. Both qualitative and quantitative evaluations on challenging image sequences demonstrate that the proposed tracking algorithm achieves more favorable performance than several state-of-the-art methods.

## 1. Introduction

As one of the fundamental problems of computer vision, visual tracking plays a critical role in advanced vision-based applications (e.g., visual surveillance, human-computer interaction, augmented reality, intelligent transportation, and context-based video compression) [1–3]. However, building a robust model-free tracker is still a challenging issue due to the difficulty arising from the appearance variability of an object of interest, which includes intrinsic appearance variability (e.g., pose variation and shape deformation) and extrinsic factors (illumination changes, camera motion, occlusions, etc.).

Typically, a complete tracking system can be divided into three main components: (1) an appearance observation model, which evaluates the likelihood of a candidate state belonging to the object model, (2) a motion model, which aims to model the states of an object over time (such as Kalman filtering and particle filtering), and (3) a search strategy for finding the most likely states in the current frame (e.g., mean shift and sliding window). In this paper, we are devoted to developing a robust appearance model.

Due to the power of subspace representation, subspace-based trackers (e.g., [4, 5]) are robust to in-plane rotation, scale change, illumination variation, and pose change. However, they are sensitive to partial occlusion caused by their underlying assumption that the error term is Gaussian distributed with small variances. This assumption does not hold for object representation when partial occlusion occurs as the noise term cannot be modeled with small variances.

An effective tracking algorithm (called L1 tracker) based on sparse representation within a particle filter framework is developed in [6]. The L1 tracker represents the tracked target by using a set of target templates and trivial templates. The target templates depict a subspace on the tracked object and the trivial templates aim to model the occlusion effectively. However, the use of trivial templates increases the number of templates significantly, which make the computational complexity of L1 tracker too high to satisfy real applications.

In [7], the authors also presented a sparse coding-based tracker by combining sparse coding and Kalman filtering and fusing the color and gradient features. To account for the variations of the tracked object during the tracking

processing, they use a template update strategy by replacing a random template of the original template library with the last tracking result. However, this simple update manner can easily introduce tracking errors when abnormal changes occur, which may cause tracking drift.

Motivated by aforementioned discussions, we propose an object tracking algorithm based on 2DPCA and MLE. Firstly, we introduce regularization into the 2DPCA reconstruction and develop an iterative algorithm to represent an object by 2DPCA bases. Secondly, the model of sparsity constrained MLE is established. Abnormal pixels in the samples will be assigned with low weights to reduce their affects on the tracking algorithm. The object tracking results are obtained by using Bayesian maximum a posteriori probability (MAP) estimation. Finally, to further reduce tracking drift, we employ a template update strategy which combines incremental subspace learning and the error matrix. This strategy adapts the template to the appearance change of the target and reduces the influence of the occluded target template as well. The experimental results show that our algorithm can achieve stable and robust performance especially when occlusion, rotation, scaling, or illumination variation occurs.

## 2. Visual Object Tracking Model Based on 2DPCA and MLE: The Theory of 2DPCA

*2.1. The Theory of 2DPCA.* Principal component analysis (PCA) is a well-established linear dimension-reduction technique, which has been widely used in many areas (such as face recognition [8]). It finds the projection directions along which the reconstruction error to the original data is minimum and projects the original data into a lower dimensional space spanned by those directions corresponding to the top eigenvalues. Recent studies demonstrate that two-dimensional principal component analysis (2DPCA) could achieve performance comparable to PCA with less computational cost [9, 10].

Given a series of image matrices  $\mathbf{Y} = [Y_1 \ Y_2 \ \cdots \ Y_d]$ , 2DPCA aims to obtain an orthogonal left-projection matrix  $\mathbf{U}$ , an orthogonal right-projection matrix  $\mathbf{V}$ , and the projection coefficients  $\mathbf{A} = [A_1 \ A_2 \ \cdots \ A_d]$  by solving the following objective function:

$$\min_{\mathbf{U}, \mathbf{V}, \mathbf{A}_i} \frac{1}{d} \sum_{i=1}^d \|\mathbf{Y}_i - \mathbf{U}\mathbf{A}_i\mathbf{V}'\|_F^2. \quad (1)$$

Then the coefficient  $\mathbf{A}_i$  can be approximated by  $\mathbf{A}_i \approx \mathbf{U}'\mathbf{Y}_i\mathbf{V}$ . We note that the underlying assumption of (1) is that the error term is Gaussian distributed with small variances. This assumption is not able to deal with partial occlusion as the error term cannot be modeled with small variances when occlusion occurs. In this paper, we propose an object tracking algorithm by using 2DPCA basis matrices and an additional MLE error matrix  $\mathbf{Y} \approx \mathbf{U}\mathbf{A}\mathbf{V}' + \mathbf{e}$ .

Let the objective function be

$$L(\mathbf{A}, \mathbf{E}) = \frac{1}{2} \|\mathbf{Y} - \mathbf{U}\mathbf{A}\mathbf{V}' - \mathbf{E}\|_F^2 + \lambda \|\mathbf{e}\|_1; \quad (2)$$

the problem is

$$\begin{aligned} \min_{\mathbf{A}, \mathbf{E}} \quad & L(\mathbf{A}, \mathbf{E}) \\ \text{s.t.} \quad & \mathbf{U}'\mathbf{U} = \mathbf{I}; \quad \mathbf{V}'\mathbf{V} = \mathbf{I}, \end{aligned} \quad (3)$$

where  $\mathbf{Y}$  denotes an observation matrix,  $\mathbf{A}$  indicates its corresponding projection coefficient, and  $\lambda$  is a regularization parameter.  $\mathbf{e}$  describes the error matrix.

*2.2. MLE Model.* The basic idea of sparse coding is to use the templates in a given dictionary  $\mathbf{T}$  to represent a testing sample  $y$  (as  $y \approx T\alpha$ ), where  $\alpha$  is sparse coding coefficient vector. Traditionally, the sparsity can be measured by L0-norm and the L0-norm minimization is an NP-hard problem. Fortunately, [11] proves that when the solution is sparse enough, L0-norm minimization is equivalent to the L1-norm minimization.

Therefore, the sparse coding problem can be defined as [12, 13]

$$\begin{aligned} \min_{\alpha} \quad & \|\alpha\|_1 \\ \text{s.t.} \quad & \|y - T\alpha\|_2^2 \leq \varepsilon, \end{aligned} \quad (4)$$

where  $\varepsilon > 0$  is a very small constant. This model shows two constraints in sparse coding: one is that  $\min_{\alpha} \|\alpha\|_1$  constrains the sparsity of represented signal; the other is that  $\|y - T\alpha\|_2^2 \leq \varepsilon$  constrains the accuracy of the represented signal [14–17].

The analysis of the two constraint terms mentioned earlier is as follows. For object tracking, the accuracy constraint is more important than the sparsity one, especially when occlusion, rotation, scaling, or illumination variation happens to the object. In that case, considering some possible abnormal changes, whether the model can accurately describe the object or not will directly determine the success or failure of tracking algorithm. Most of current algorithms are presented under the assumption that the sparse coding residual  $\mathbf{e} = y - T\hat{\alpha}$  follows the Gaussian distribution. In practice, however, this assumption is limited when abnormal changes happen which will inevitably lead to the failure of tracking algorithm.

In sparsity constraints, though L1-norm minimization is more efficient than the L0-norm minimization, the fact is that the L1-norm minimization programming is still very time consuming. Object tracking algorithms are different from face recognition algorithms in that face recognition algorithms do not demand fast processing speed in a sample training process, while in object tracking, slow processing speed will directly affect the practical value of the object tracking algorithm. In that case, the introduction of L1-norm minimization into the field of object tracking would greatly reduce the performance of tracking algorithms.

We note that the tracking accuracy and speed are two important aspects for evaluating the performance of object tracking algorithms. Therefore, in this paper, we develop an MLE-based model that improves the traditional sparse coding model from the two aspects and then apply it to achieve an effective and efficient tracker.

In the field of object tracking, accuracy is the most important issue. Hence, at first, we need to improve the accuracy constraint term in the traditional sparse coding model.

When the reconstruction error  $\mathbf{e} = \mathbf{y} - T\hat{\alpha}$  follows the Gaussian distribution, the traditional sparse coding solution can be written as

$$\hat{\alpha} = \underset{\alpha}{\operatorname{argmin}} \left\{ \|\mathbf{y} - T\alpha\|_2^2 + \lambda \|\alpha\|_1 \right\}, \quad (5)$$

where  $\lambda$  is a regularization parameter. For object tracking, the dictionary  $\mathbf{T} = [t_1, t_2, \dots, t_n] \in \mathbb{R}^{d \times n}$  consists of  $n$  templates and forms the object template library. consider  $t_i \in \mathbb{R}^d$ ,  $d \gg n$ . In our experiments, we make  $n = 20$  and the object template size  $32 \times 32$ ; that is,  $d = 1024$ . The result image block in current frame is denoted as  $\mathbf{y}$ ,  $\mathbf{y} \in \mathbb{R}^d$ .  $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_n)^t \in \mathbb{R}^n$  denotes the coefficient vector of sparse coding. Equation (5) is obviously a minimum variance estimation problem with sparsity constraint. When object's reconstruction error  $\mathbf{e} = \mathbf{y} - T\hat{\alpha}$  follows the Gaussian distribution, the solution of (5) is the maximum likelihood estimation.

However, in practical applications, when the object suffers from occlusion, rotation change, scale change, or illumination variation, the reconstruction errors  $\mathbf{e}$  of abnormal pixels will not follow the Gaussian distribution. In that case, these algorithms may not track the object accurately. Therefore, we need to build a more adaptive object representing model.

First, we rewrite the dictionary  $\mathbf{T}$  as  $\mathbf{T} = [r_1; r_2; \dots; r_d]$ , where row vector  $r_j \in \mathbb{R}^n$ ,  $j = 1, 2, \dots, d$ , is the  $j$ th row of  $\mathbf{T}$ . Meanwhile, we rewrite tracking result image block  $\mathbf{y}$  as  $\mathbf{y} = [y_1; y_2; \dots; y_d]$ , where  $y_j$ ,  $j = 1, 2, \dots, d$ , is the  $j$ th pixel of  $\mathbf{y}$ . In that case, the reconstruction error  $\mathbf{e} = \mathbf{y} - T\hat{\alpha} = [e_1; e_2; \dots; e_d]$ , where  $e_j = y_j - r_j \hat{\alpha}$ ,  $j = 1, 2, \dots, d$ , is the  $j$ th pixel's reconstruction error.

Assume that  $e_1, e_2, \dots, e_n$  are independently and identically distributed according to a certain probability density function  $p_\theta(e_j)$ , where  $\theta$  denotes the parameter set that characterizes the distribution. Then the likelihood function would be  $L_\theta(e_1, e_2, \dots, e_d) = \prod_{j=1}^d p_\theta(e_j)$ , and MLE aims to maximize this likelihood function or, equivalently, minimize the objective function:  $-\ln L_\theta(e_1, e_2, \dots, e_d) = \sum_{j=1}^d f_\theta(e_j)$ , where  $f_\theta(e_j) = -\ln p_\theta(e_j)$ , to simplify the computation.

Taking into account the sparsity constraint of  $\alpha$ , the MLE of  $\alpha$  can be formulated as the following minimization:

$$\hat{\alpha} = \underset{\alpha}{\operatorname{argmin}} \left\{ \sum_{j=1}^d f_\theta(e_j) + \lambda \|\alpha\|_1 \right\}. \quad (6)$$

According to [6], formula (6) can be converted into weighted sparse coding problem

$$\hat{\alpha} = \underset{\alpha}{\operatorname{argmin}} \left\{ \|W^{1/2}(\mathbf{y} - T\alpha)\|_2^2 + \lambda \|\alpha\|_1 \right\}, \quad (7)$$

where  $W$  is a diagonal matrix with diagonal elements as follows:

$$W_{j,j} = \frac{\exp(\mu\sigma - \mu e_j^2)}{(1 + \exp(\mu\sigma - \mu e_j^2))}, \quad (8)$$

which also stands for the  $j$ th pixel's weight value.  $\mu$  and  $\sigma$  are positive constants. If we make  $W_{j,j} = 2$ , then the model would be the traditional sparse coding problem. Hence, we can see that formula (7) is more adaptive than (3).

In this study, we choose it as the weight function

$$W_{j,j} = \frac{1}{1 + 1/\exp(-\beta e_j)}, \quad (9)$$

where  $\beta$  is a scale factor (we choose  $\beta = 10$  in our experiments). The physical meaning of  $W_{j,j}$  is to allocate smaller weights to those pixels with bigger residuals (probably abnormal pixels) and allocate bigger weights to pixels with smaller residuals. By setting a reasonable weight threshold, we can get rid of those abnormal pixels lower than the threshold and do further sparse coding. In that case, we can effectively reduce the effect of abnormal pixels and therefore achieve good performance during the tracking processing.

From (9), we can see that the weight value  $W_{j,j}$  is bounded between 0 and 1 which makes sure that even the pixels with very small residuals would not have too large weight values. This would guarantee the stability of the algorithm.

### 3. Bayesian MAP Estimation

We can regard object tracking as a hidden state variables' Bayesian MAP estimation problem in the Hidden Markov model; that is, with a set of observed samples  $Y_t = \{y_1, y_2, \dots, y_t\}$ , we can estimate the hidden state variable  $x_t$  using Bayesian MAP theory.

According to the Bayesian theory,

$$p(x_t | Y_t) \propto p(y_t | x_t) \int p(x_t | x_{t-1}) p(x_{t-1} | Y_{t-1}) dx_{t-1}, \quad (10)$$

where  $p(x_t | x_{t-1})$  stands for a state transition model for two consecutive frames and  $p(y_t | x_t)$  stands for an observation likelihood model. We can obtain the object's best state in  $t$ th frame through maximum posterior probability estimation; that is,

$$\hat{x}_t = \underset{x_t^l}{\operatorname{argmax}} p(x_t^l | Y_t), \quad l = 1, 2, \dots, N, \quad (11)$$

where  $x_t^l$  stands for the  $l$ th sample of state variable  $x_t$  in  $t$ th frame. In this paper, we choose  $N = 400$ .

**3.1. State Transition Model.** We choose object's motion affine transformation parameters as state variable  $x_t = \{x_t, y_t, \theta_t, S_t, \alpha_t, \phi_t\}$ , where  $x_t$  and  $y_t$ , respectively, represent the  $x$ -direction and  $y$ -direction translation of the object in  $t$ th



FIGURE 1: Tacking results of test video “Car4.”



FIGURE 2: Tacking results of test video “David.”

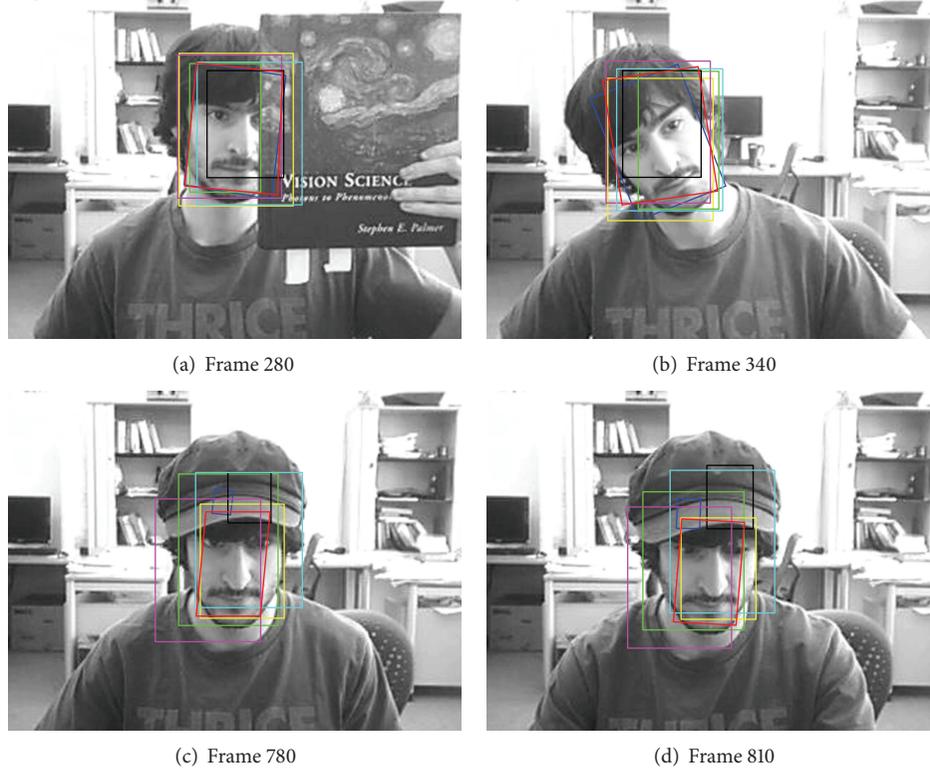


FIGURE 3: Tacking results of test video “Faceocc2.”

frame,  $\theta_t$  stands for the rotation angle,  $S_t$  represents the scale change,  $\alpha_t$  stands for the aspect ratio, and  $\phi_t$  stands for the direction of tilt.

We assume that the state transition model follows the Gaussian distribution; that is,

$$p(x_t | x_{t-1}) = N(x_t; x_{t-1}, \Psi), \quad (12)$$

where  $\Psi$  is a diagonal matrix whose diagonal elements are motion affine parameter’s variation  $\sigma_x^2, \sigma_y^2, \sigma_\theta^2, \sigma_S^2, \sigma_\alpha^2, \sigma_\phi^2$ .

**3.2. Observation Likelihood Model.** We use object’s reconstruction error to build observation likelihood model; that is,

$$p(y_t | x_t) = \prod_{j=1,2,\dots,d} \mathbb{N}(e_j^t, \mu, \sigma^2), \quad (13)$$

where  $\mathbb{N}(\cdot)$  means Gaussian distribution,  $\mu$  and  $\sigma^2$ , respectively, represent the mean and variation of Gaussian distribution,  $d$  stands for the number of pixels of an object template, and  $e_j^t = \|y_j^t - \Phi_j^t \hat{y}_j^t\|_2$  stands for the reconstruction error of  $j$ th pixel of object templates in  $t$ th frame.

**3.3. Templates Updating.** To consider that the appearance of the target may change during the tracking processing, it is necessary to dynamically update the template library.

In this paper, we use a method named “Half Updating Strategy” to update the templates. We take the tracking results

TABLE 1: The description of test videos.

Name of test videos	Number of frames	Video description
Car4	659	Illumination variation and scale variation
David	462	Illumination variation, in-plane rotation, and off-plane rotation
Faceocc2	819	Partly occlusion, in-plane rotation, and off-plane rotation

of first  $n$  frames as the initial templates, and from  $(n + 1)$ th frame on, we use the algorithm mentioned earlier to obtain and save the tracking results. During this process, if the result image block has equal to 50% abnormal pixels, then we do not update the tracker. When we have  $n/2$  tracking results, that is, half of the number of initial templates, we replace the first  $n/2$  templates in original template library with the newly accumulated  $n/2$  tracking results. Then a “Half Updating” is finished.

## 4. Experimental Results and Analysis

In order to evaluate the performance of our tracker, we conduct experiments on three challenging image sequences (Table 1 and Figures 1, 2, and 3). These sequences cover most challenging situations in object tracking: occlusion,

motion blur, in-plane and out-of-plane rotation, large illumination change, scale variation, and complex background. For comparison, we run six state-of-the-art algorithms with the same initial position of the target. These algorithms are the Frag tracking [18], IVT tracking [19], MIL tracking [20], L1 tracking [6], PN tracking [21], and VTD tracking [22] methods. We present some representative results in this section.

## 5. Conclusions/Outlook

This paper presents a robust tracking algorithm via 2DPCA and MLE. In this work, we represent the tracked object by using 2DPCA bases and an MLE error matrix. With the proposed model, we can remove the abnormal pixels and thus reduce the effect of abnormal pixels on tracking algorithms. We take the object's reconstruction error into the Bayesian maximum posterior probability estimation framework and design a stable and robust tracker. Then, we explicitly take partial occlusion and misalignment into account for appearance model update and object tracking. Experiments on challenging video clips show that our tracking algorithm performs better than several state-of-the-art algorithms. Our future work will be the generalization of our representation model into other related fields.

## Acknowledgments

This research described in this paper was supported by the Fundamental Research Funds for the Central Universities (DC110321, DC120101132, and DC120101131). This work was supported by Project of Liaoning Provincial Department of Education (L2012476, and L2010094). This work was supported by National Natural Science Foundation of China (61172058).

## References

- [1] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: a survey," *ACM Computing Surveys*, vol. 38, no. 4, pp. 229–240, 2006.
- [2] M. X. Jiang, Z. J. Shao, and H. Y. Wang, "Real-time object tracking algorithm with cameras mounted on moving platforms," *International Journal of Image and Graphics*, vol. 12, no. 3, Article ID 1250020, 2012.
- [3] M. Jiang, M. Li, and H. Wang, "A robust combined algorithm of object tracking based on moving object detection," in *Proceedings of the International Conference on Intelligent Control and Information Processing (ICICIP '10)*, pp. 619–622, Dalian, China, July 2010.
- [4] L. Zhang, P. Zhu, Q. Hu, and D. Zhang, "A linear subspace learning approach via sparse coding," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV '11)*, pp. 755–761, Barcelona, Spain, November 2011.
- [5] M. X. Jiang, M. Li, and H. Y. Wang, "Object tracking algorithm based on projection matrix," *Journal of Convergence Information Technology*, vol. 7, pp. 209–217, 2012.
- [6] X. Mei and H. Ling, "Robust visual tracking and vehicle classification via sparse representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 11, pp. 2259–2272, 2011.
- [7] Z. Han, J. Jiao, B. Zhang, Q. Ye, and J. Liu, "Visual object tracking via sample-based Adaptive Sparse Representation (AdaSR)," *Pattern Recognition*, vol. 44, no. 9, pp. 2170–2183, 2011.
- [8] M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of Cognitive Neuroscience*, vol. 3, no. 1, pp. 71–86, 1991.
- [9] J. H. Yin, C. Y. Fu, and J. K. Hu, "Using incremental subspace and contour template for object tracking," *Journal of Network and Computer Applications*, vol. 35, pp. 1740–1748, 2012.
- [10] D. Wang, H. Lu, and X. Li, "Two dimensional principal components of natural images and its application," *Neurocomputing*, vol. 74, no. 17, pp. 2745–2753, 2011.
- [11] D. L. Donoho, "For most large underdetermined systems of linear equations the minimal  $\ell_1$ -norm solution is also the sparsest solution," *Communications on Pure and Applied Mathematics*, vol. 59, no. 6, pp. 797–829, 2006.
- [12] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 210–227, 2009.
- [13] A. Wagner, J. Wright, A. Ganesh, Z. Zhou, H. Mobahi, and Y. Ma, "Toward a practical face recognition system: robust alignment and illumination by sparse representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 2, pp. 372–386, 2012.
- [14] Y. Meng, L. Zhang, J. Yang, and D. Zhang, "Robust sparse coding for face recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '11)*, pp. 625–632, Colorado Springs, Colo, USA, June 2011.
- [15] L. Zhang, M. Yang, and X. Feng, "Sparse representation or collaborative representation: which helps face recognition?" in *Proceedings of the IEEE International Conference on Computer Vision (ICCV '11)*, pp. 471–478, Barcelona, Spain, November 2011.
- [16] C. Chiang, C. Duan, S. Lai, and S. Chang, "Learning component-level sparse representation using histogram information for image classification," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV '11)*, pp. 1519–1526, Barcelona, Spain, November 2011.
- [17] W. Dong, X. Li, L. Zhang, and G. Shi, "Sparsity-based image denoising via dictionary learning and structural clustering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '11)*, pp. 457–464, Barcelona, Spain, November 2011.
- [18] A. Adam, E. Rivlin, and I. Shimshoni, "Robust fragments-based tracking using the integral histogram," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '06)*, pp. 798–805, New York, USA, June 2006.
- [19] D. A. Ross, J. Lim, R. Lin, and M. Yang, "Incremental learning for robust visual tracking," *International Journal of Computer Vision*, vol. 77, no. 1–3, pp. 125–141, 2008.
- [20] B. Babenko, M. H. Yang, and S. Belongie, "Visual tracking with online multiple instance learning," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPR '09)*, pp. 983–990, Miami, Fla, USA, June 2009.

- [21] Z. Kalal, J. Matas, and K. Mikolajczyk, "P-N learning: bootstrapping binary classifiers by structural constraints," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '10)*, pp. 49–56, San Francisco, Calif, USA, June 2010.
- [22] J. Kwon and K. M. Lee, "Visual tracking decomposition," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '10)*, pp. 1269–1276, San Francisco, Calif, USA, June 2010.

## Research Article

# Practical Speech Emotion Recognition Based on Online Learning: From Acted Data to Elicited Data

**Chengwei Huang, Ruiyu Liang, Qingyun Wang, Ji Xi, Cheng Zha, and Li Zhao**

*School of Information Science and Engineering, Southeast University, Nanjing 210096, China*

Correspondence should be addressed to Chengwei Huang; [huangcwx@126.com](mailto:huangcwx@126.com)

Received 7 March 2013; Revised 26 May 2013; Accepted 4 June 2013

Academic Editor: Saeed Balochian

Copyright © 2013 Chengwei Huang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We study the cross-database speech emotion recognition based on online learning. How to apply a classifier trained on acted data to naturalistic data, such as elicited data, remains a major challenge in today's speech emotion recognition system. We introduce three types of different data sources: first, a basic speech emotion dataset which is collected from acted speech by professional actors and actresses; second, a speaker-independent data set which contains a large number of speakers; third, an elicited speech data set collected from a cognitive task. Acoustic features are extracted from emotional utterances and evaluated by using maximal information coefficient (MIC). A baseline valence and arousal classifier is designed based on Gaussian mixture models. Online training module is implemented by using AdaBoost. While the offline recognizer is trained on the acted data, the online testing data includes the speaker-independent data and the elicited data. Experimental results show that by introducing the online learning module our speech emotion recognition system can be better adapted to new data, which is an important character in real world applications.

## 1. Introduction

The state-of-the-art speech emotion recognition (SER) system is largely dependent on its training data. Emotional vocal behavior is personality dependent, situation dependent, and language dependent. Therefore, emotional models trained from a specific database may not fit to other databases. To solve this problem, we introduce an online learning framework to the SER system. Online speech data is used to retrain and to improve the classifier. Adopting the online learning framework, we may better adapt our SER system to different speakers and different data sources.

Many achievements have been reported on the acted speech emotion databases [1–3]. Tawari and Trivedi [4] considered the role of context and detected seven emotions on the Berlin Emotional Database [5]. Ververidis and Kotropoulos [6] studied gender-based speech emotion recognition system for five different emotional states. A number of machine learning algorithms have been studied in SER, using acted emotional data. Only recently the need of using naturalistic data has been pointed out. Several naturalistic speech emotion databases have been developed, such as AIBO emotional

speech database [7] and VAM database [8]. Many researchers notice that real world data plays a key role in the SER system [9], and the model trained on the acted data does not fit very well on the naturalistic data.

Incremental learning may provide us a good solution to solve this problem under an online learning framework. The pretrained models on the acted data may be updated using very few online data. Since the naturalistic emotion data is very difficult to collect, acted speech data still plays an important role, especially in studying rare emotion types, such as fear-type emotion [1], confidence, and anxiety [10]. By using incremental learning we can make use of the available acted databases as a baseline recognizer and then retrain the classifier online for specific purposes.

Many successful algorithms have been proposed for incremental learning, such as Learning++ [11] and Bagging++ [12]. Incremental learning algorithms may be classified into two categories. In the first category, a single classifier is updated by reestimating its parameters. This type of learning algorithms is dependent on the specific classifier, such as the incremental learning algorithm for support vector machine

proposed by Xiao et al. [13]. The techniques used in such parameter estimation may not be generalized. In the second category, the incremental learning algorithm is not dependent on a specific type of classifiers. Multiple classifiers are created and combined by a certain fusion rule, such as majority vote. Boosting is a typical type of algorithms that fall into the second category. By creating weak classifiers using selected data, we may add new training data to the learning procedure and gradually adapt the SER system in an online environment.

In this paper we explore the possibility of transferring pretrained SER system from acted data to more naturalistic data in an online learning framework. Section 2 describes our acted data and elicited data. Section 3 provides acoustic analysis of emotional features. In Section 4, we introduce our speech emotion recognizer and the online learning methodology. Finally, in Section 5, we provide the experimental results, which show that combining the acted data and the elicited data using online learning brings us the best result.

## 2. Three Types of Data Sources

In this paper we use three types of data sources to validate our SER system: (i) acted basic emotion database, (ii) speaker-independent emotion database, and (iii) elicited emotion database.

The first database contains the basic emotions, including happiness, anger, surprise, sadness, fear, and neutrality. The emotional speech is recorded by professional actors and actresses, six males and six females. This acted database may be used as a standard training dataset for our baseline recognizer. However, in real world applications the naturalistic emotional speech is different from the acted speech.

The second database is designed for speaker-independent test, which includes fifty-one different speakers. Other than a large number of speakers, a special type of emotion is considered, namely, fidgetiness. Fidgetiness is an important type of emotion in cognitive related tasks. It may be induced by repeated work, environmental noise, and stress. The second database contains five emotions, as shown in Table 1. This database may be used for testing the ability of speaker adaptation. When using training data from the first database, it is challenging to test our SER system on the second database, due to many unknown speakers.

The third database contains elicited speech in a cognitive task, as shown in Table 2. The first row shows the emotion types collected in our experiments, such as fidgetiness, confidence, and tiredness. The second row is the speaker number related to each type of emotion. The third row is the male and female proportion in the emotion data. The last row is the number of utterances in each emotion class. The data is collected locally in our lab. We carried out a cognitive experiment and collected the emotional speech related to cognitive performance. Subject was required to work on a set of math calculations and to report the results orally. During the cognitive task the speech signals were recorded and annotated with emotional labels.

In the third database, “correct answer” or “false answer” labels are marked on each utterance in the oral report by

TABLE 1: The Speaker-independent emotion dataset.

Emotion type	Happiness	Anger	Fidgetiness	Sadness	Neutrality
Speaker number	51	51	51	51	51
Male/female	23/28	23/28	23/28	23/28	23/28
Utterance size	2200	2200	2200	2200	2200

TABLE 2: The Elicited Emotion Dataset.

Emotion type	Confidence	Tiredness	Fidgetiness	Happiness	Neutrality
Speaker number	6	6	6	6	6
Male/female	3/3	3/3	3/3	3/3	3/3
Utterance size	1200	1200	1200	1200	1200

the listeners who have not participated in the eliciting experiment. Therefore we may calculate the percentage of false answers in the negative emotion samples and the percentage of negative emotion in the “false answer” samples. Results show that the proportion of the mistake made in the math calculation is higher with the presence of negative emotions, as shown in Figures 1 and 2. The purpose of this database is to study the cognitive related emotions in speech. The analysis shows the dependency between the mistakes made in the math calculation and the negative emotions in the speech.

## 3. Feature Analysis

*3.1. Acoustic Feature Extraction.* Emotional information is hidden in the speech signals. Unlike the linguistic information, it is difficult to find the related acoustic features. Therefore feature analysis and selection are very important steps in building an SER system.

We selected typical utterances to study the feature variance caused by emotional change, as shown in Figures 3, 4, 5, 6, 7, 8, 9, 10, and 11. To better reflect the change caused by emotional information, we fix the context of these utterances.

The utterances shown in the figures are recorded from the same speaker. By comparing the utterances under different emotional state from the same speaker, we can exclude the influence brought by different speaking habits and personalities. It reveals the changes in the acoustic features caused only by the emotional information.

We induced three types of practical emotions from a cognitive task, namely, fidgetiness, confidence, and tiredness. We also studied the basic emotions, like happiness, anger, surprise, sadness, and fear. The intensity feature and the pitch contour are extracted and demonstrated in Figure 3 through Figure 11.

The first syllable is not normal speech under the fear emotional state. The pitch feature is missing, and it is whispered speech under the emotional state of fear. Under the tiredness emotion state, the pitch contour is low and flat, which is quite distinguishable from other emotion states.



FIGURE 1: The percentage of negative emotions when mistake occurs in the cognitive task.



FIGURE 2: The percentage of correct answers and false answers when negative emotion occurs in the cognitive task.

In the neutral speech, the pitch contour is also flat, but at the end of the sentence the pitch frequency increases. Comparing speaking, the pitch frequency is not consistent at the end of the sentence. Under the sadness emotion state, the pitch contour is smooth and decreases at the end of the sentence. Furthermore, in the happiness sample, the variance of the pitch frequency is higher. The pitch frequency also increases in the confidence and surprise samples.

We also notice that under the angry emotion state the variance of the intensity is lower and the intensity contour is smooth. However, in the sadness sample, the variance of the intensity is higher. Sadness and tiredness may have caused longer time duration and a lower speech rate, while fidgetiness and anger may have caused a higher speech rate.

Quantitative statistical analysis is shown in Figure 12. Pitch and formants features are compared under various emotional states.

For modeling and recognition purposes, 481 dimensions of acoustic features are constructed. Statistic functions over

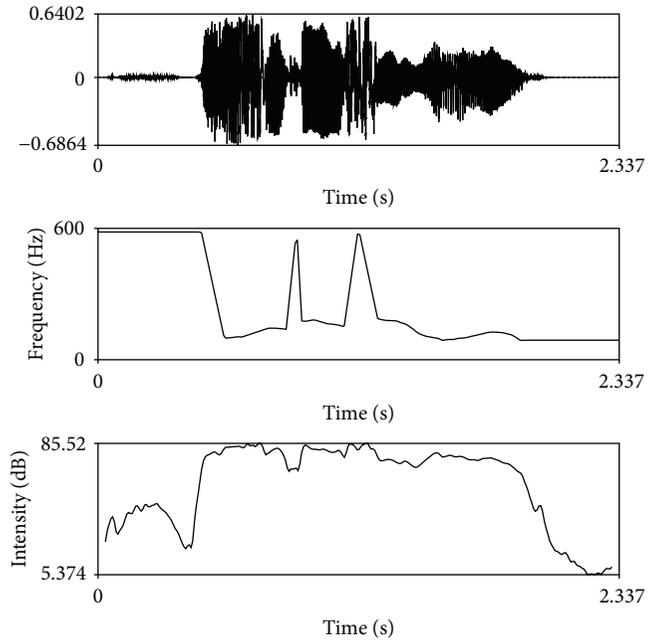


FIGURE 3: Intensity and pitch contour of happiness.

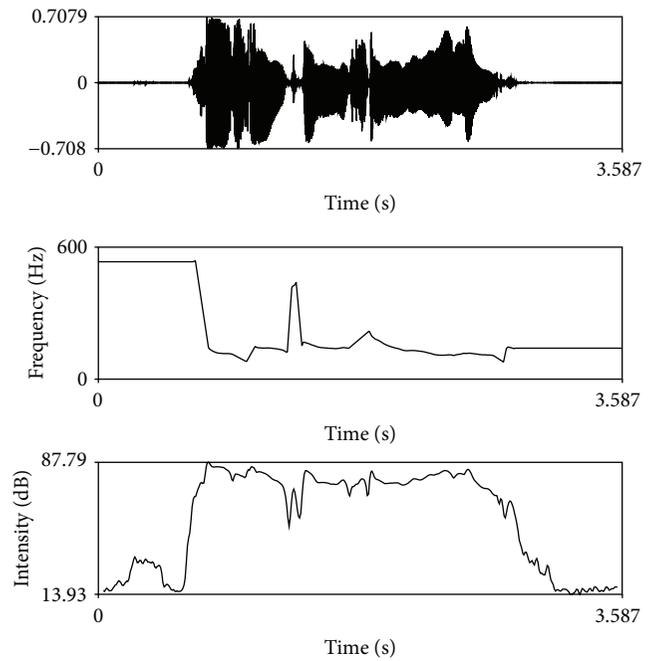


FIGURE 4: Intensity and pitch contour of sadness.

the entire utterance, such as maximum, minimum, mean, range, are applied to the basic speech features, as listed below. “d” stands for difference and “d<sup>2</sup>” stands for the second order of difference.

Feature 1–6: mean, maximum, minimum, median, range, and variance of Short-time Energy (SE).

Feature 7–18: mean, maximum, minimum, median, range, and variance of dSE and d<sup>2</sup>SE.

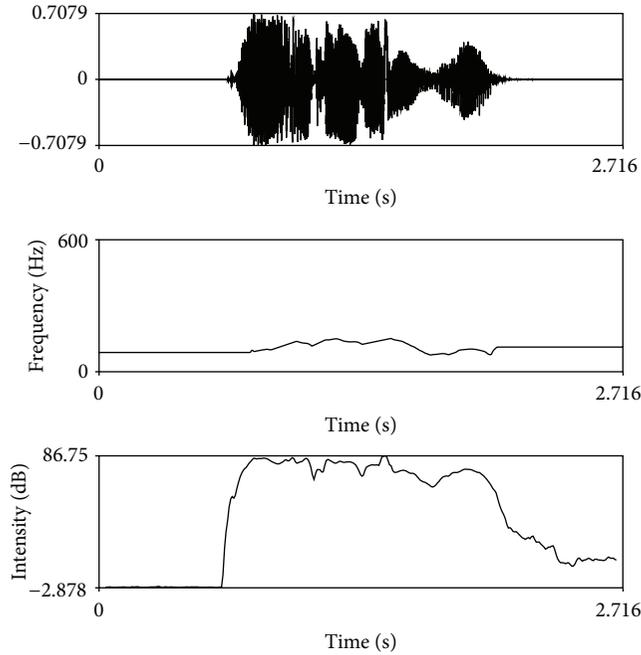


FIGURE 5: Intensity and pitch contour of fidgetiness.

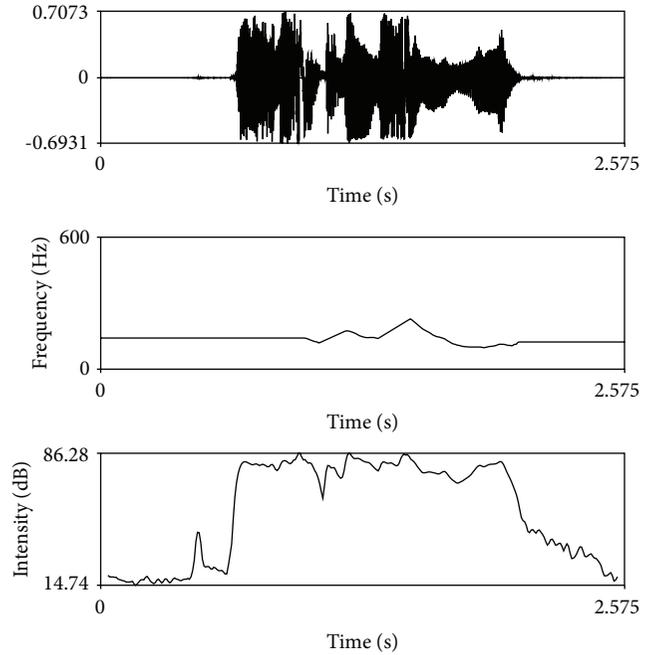


FIGURE 7: Intensity and pitch contour of fear.

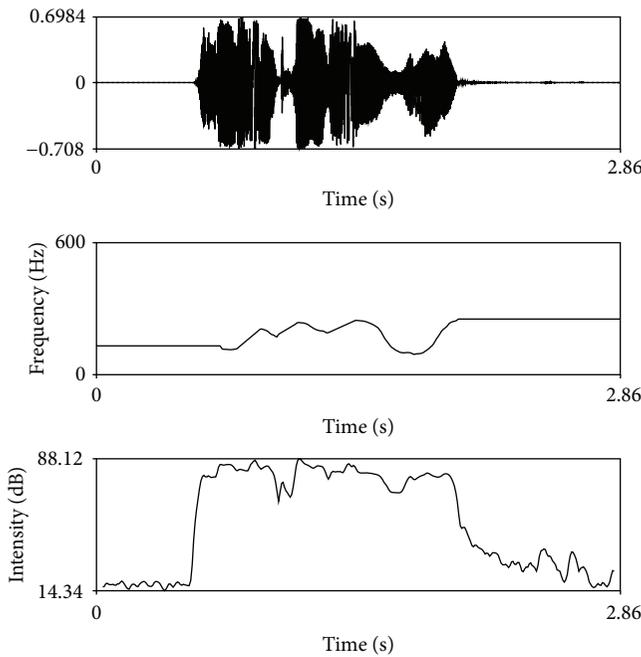


FIGURE 6: Intensity and pitch contour of surprise.

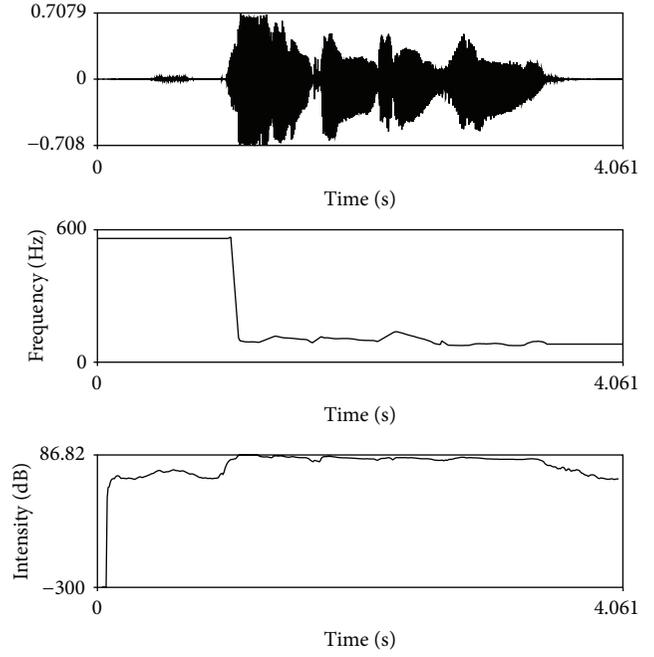


FIGURE 8: Intensity and pitch contour of tiredness.

Feature 19–24: mean, maximum, minimum, median, range, and variance of pitch frequency ( $F_0$ ).

Feature 25–36: mean, maximum, minimum, median, range, and variance of  $dF_0$  and  $d^2F_0$ .

Feature 37–42: mean, maximum, minimum, median, range, and variance of Zero-Crossing Rate (ZCR).

Feature 43–54: mean, maximum, minimum, median, range and variance of  $dZCR$  and  $d^2ZCR$ .

Feature 55: speech rate (SR).

Feature 56–57: Pitch Jitter1 (PJ1), Pitch Jitter2 (PJ2).

Feature 58–61: 0–250 Hz Energy Ratio (ER), 0–650 Hz ER, and 4 kHz above ER and Energy Shimmer (ESH).

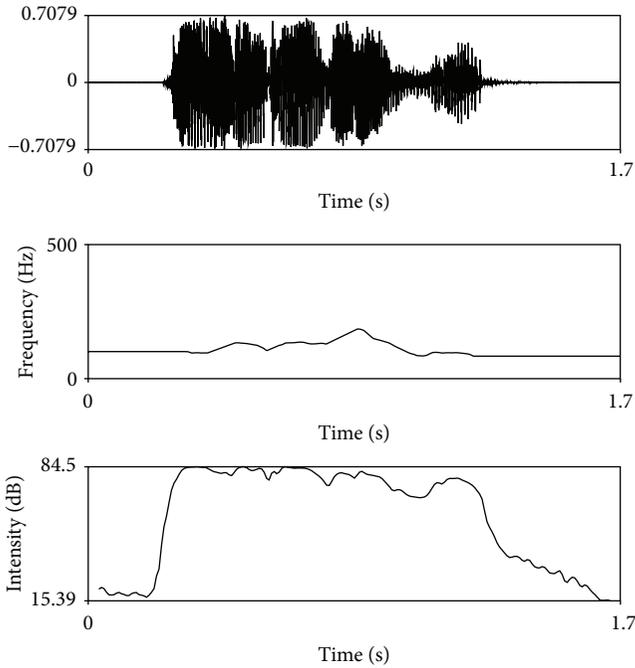


FIGURE 9: Intensity and pitch contour of anger.

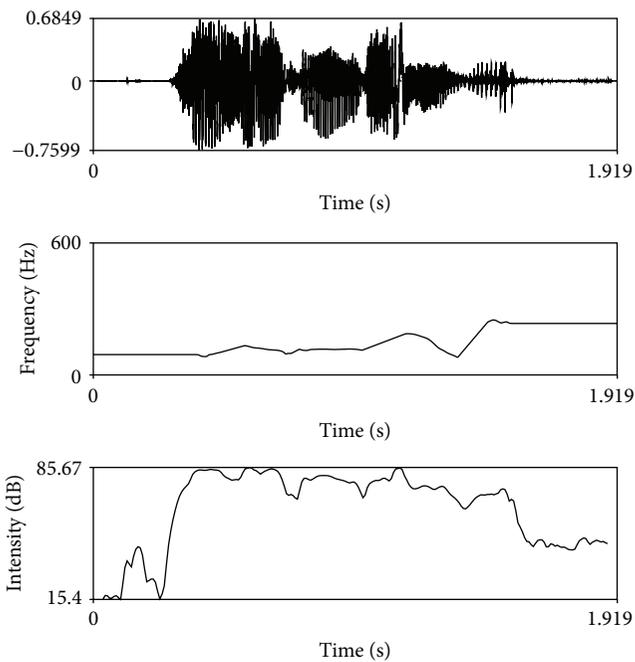


FIGURE 10: Intensity and pitch contour of neutrality.

Feature 62–65: Voiced Frames (VF), Unvoiced Frames (UF), UF/VF, and VF/(UF+VF).

Feature 66–69: Voiced Segments (VS), Unvoiced Segments (US), US/VS, and VS/(US+VS).

Feature 70-71: Maximum Voiced Duration (MVD), Maximum Unvoiced Duration (MUD).

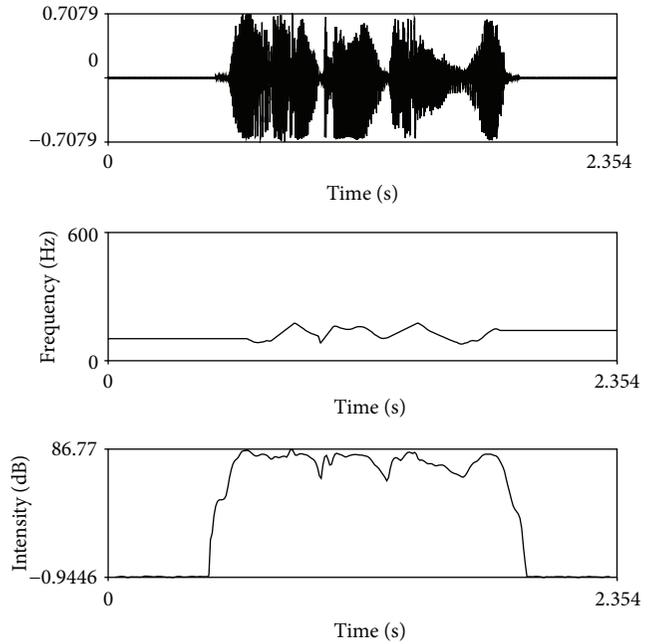


FIGURE 11: Intensity and pitch contour of confidence.

Feature 72–77: mean, maximum, minimum, median, range, and variance of Harmonic-to-Noise Ratio (HNR).

Feature 78–95: mean, maximum, minimum, median, range, and variance of HNR (0–400 Hz, 400–2000 Hz, and 2000–5000 Hz).

Feature 96–119: mean, maximum, minimum, median, range, and variance of 1st formant frequency (F1), 2nd formant frequency (F2), 3rd formant frequency (F3), and 4th formant frequency (F4).

Feature 120–143: mean, maximum, minimum, median, range, and variance of dF1, dF2, dF3, and dF4.

Feature 144–167: mean, maximum, minimum, median, range, and variance of d<sup>2</sup>F1, d<sup>2</sup>F2, d<sup>2</sup>F3, and d<sup>2</sup>F4.

Feature 168–171: Jitter1 of F1, F2, F3, and F4.

Feature 172–175: Jitter2 of F1, F2, F3, and F4.

Feature 176–199: mean, maximum, minimum, median, range, and variance of F1, F2, F3, and F4 Bandwidth.

Feature 200–223: mean, maximum, minimum, median, range, and variance of dF1 Bandwidth, dF2 Bandwidth, dF3 Bandwidth, and dF4 Bandwidth.

Feature 224–247 mean, maximum, minimum, median, range, and variance of d<sup>2</sup>F1 Bandwidth, d<sup>2</sup>F2 Bandwidth, d<sup>2</sup>F3 Bandwidth and d<sup>2</sup>F4 Bandwidth.

Feature 248–325: mean, maximum, minimum, median, range, and variance of MFCC (0–12th-order).

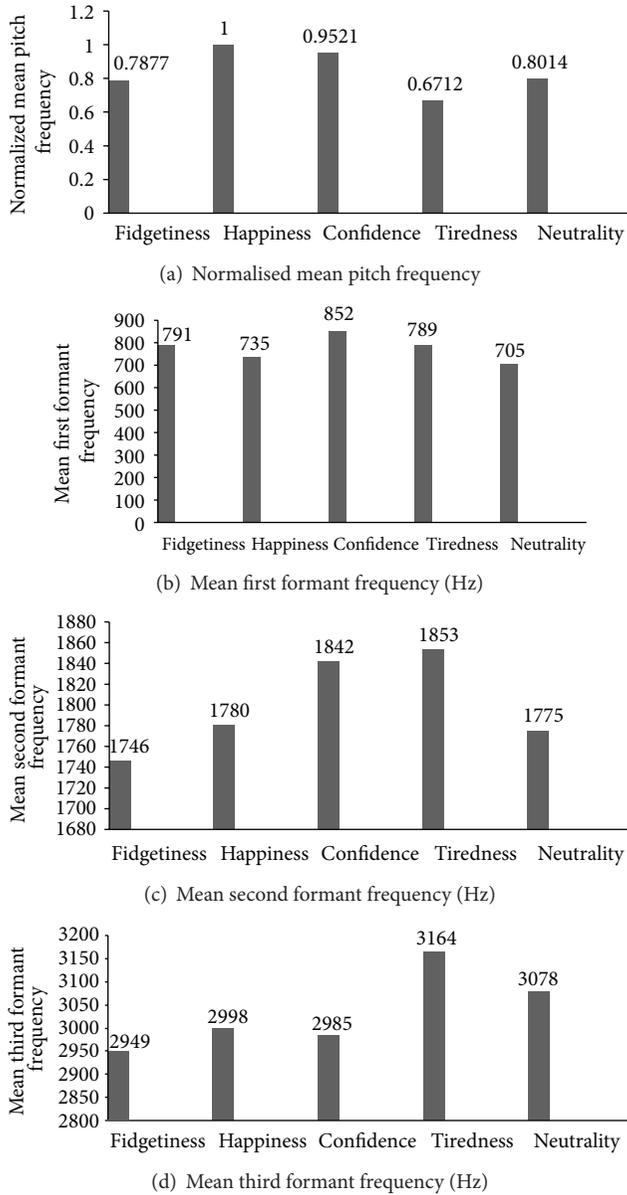


FIGURE 12: Feature distribution over various emotional states.

Feature 326–403: mean, maximum, minimum, median, range, and variance of dMFCC (0–12th-order).

Feature 404–481: mean, maximum, minimum, median, range, and variance of d2MFCC (0–12th-order).

**3.2. Feature Selection Based on MIC.** In this section we introduce the feature selection algorithm in our speech emotion classifier. Feature selection algorithms may be roughly classified into two groups, namely, “wrapper” and “filter.” Algorithms in the former group are dependent on the specific classifiers, such as sequential forward selection (SFS). The final selection result is dependent on a specific classifier. If we replace the specific classifier, the results will change. In the second group, feature selection is done by a certain evaluation criteria, such as Fisher Discriminant Ratio (FDR). The feature

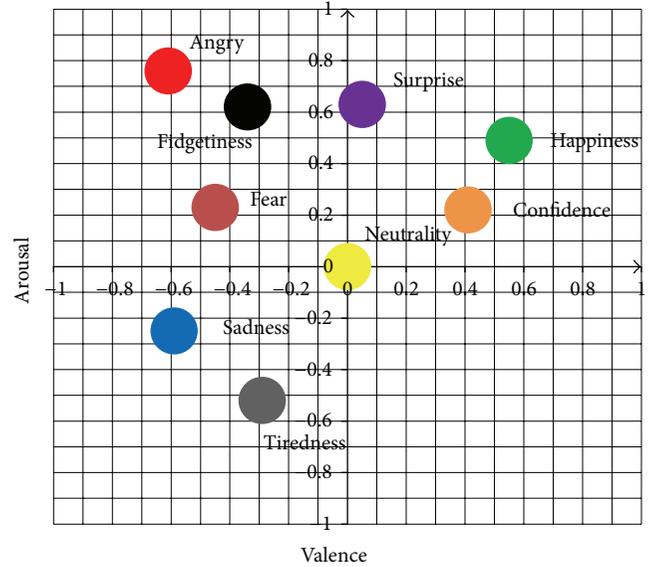


FIGURE 13: The arousal and the valence dimensions of emotions.

selection result achieved in this type of method is not dependent on specific classifiers and bears a better generality across different databases.

Maximal information coefficient (MIC) based feature selection algorithm falls into the second group. MIC is a new statistic tool that measures linear and nonlinear relationships between paired variables, invented by Reshef et al. [14].

MIC is based on the idea that if a relationship exists between two variables, then a grid can be drawn on the scatterplot of the two variables that partitions the data to encapsulate that relationship [14]. We may calculate the MIC of a certain acoustic feature and the emotional state by exploring all possible grids on the two variables. First, we compute for every pair of integers  $(x, y)$  that largest possible mutual information achieved by any  $x$ -by- $y$  grid [14]. Second, for a fair comparison we normalize these MIC values between all acoustic features and the emotional state. Detailed study of MIC may be found in [14].

Since MIC can treat linear and nonlinear associations at the same time, we do not need to make any assumption on the distribution of our original features. Therefore it is especially suitable for evaluating a large number of emotional features. Based on a large number of basic features as described in Section 3.1, we apply MIC to measure the contribution of these features in correlation with emotion states. Finally a subset of features is selected for our emotion classifier.

## 4. Recognition Methodology

**4.1. Baseline GMM Classifier.** The Gaussian mixture model (GMM) based classifier is the state-of-the-art recognition method in speaker and language identification. In this paper we built the baseline classifier using Gaussian mixture model, and we may compare the baseline classifier with the online learning method.

GMM may be defined by the sum of several Gaussian distributions:

$$p(\mathbf{X}_t | \lambda) = \sum_{i=1}^M a_i b_i(\mathbf{X}_t), \quad (1)$$

where  $\mathbf{X}_t$  is a  $D$ -dimension random vector,  $b_i(\mathbf{X}_t)$  is the  $i$ th member of Gaussian distribution,  $t$  is the index of utterance sample,  $a_i$  is the mixture weight, and  $M$  is the number of Gaussian mixture members. Each member is a  $D$ -dimension variable which follows the Gaussian distribution with the mean  $\mathbf{U}_i$  and the covariance  $\Sigma_i$ :

$$b_i(\mathbf{X}_t) = \frac{1}{(2\pi)^{D/2} |\Sigma_i|^{1/2}} \exp \left\{ -\frac{1}{2} (\mathbf{X}_t - \mathbf{U}_i)^T \Sigma_i^{-1} (\mathbf{X}_t - \mathbf{U}_i) \right\}. \quad (2)$$

Note that

$$\sum_{i=1}^M a_i = 1. \quad (3)$$

Emotion classification can be done by maximizing the posterior probability:

$$\text{EmotionLabel} = \underset{k}{\operatorname{argmax}} (p(\mathbf{X}_t | \lambda_k)). \quad (4)$$

Expectation Maximization (EM) is adopted for GMM parameter estimation [15]:

$$\begin{aligned} a_m^i &= \frac{\sum_{t=1}^T \gamma_{tm}^i}{\sum_{t=1}^T \sum_{m=1}^M \gamma_{tm}^i}, \\ \mathbf{U}_m^i &= \frac{\sum_{t=1}^T \gamma_{tm}^i \mathbf{X}_t}{\sum_{t=1}^T \gamma_{tm}^i}, \\ \Sigma_m^i &= \frac{\sum_{t=1}^T \gamma_{tm}^i (\mathbf{X}_t - \mathbf{U}_m^i) (\mathbf{X}_t - \mathbf{U}_m^i)^T}{\sum_{t=1}^T \gamma_{tm}^i}, \\ \gamma_{tm}^i &= \frac{a_m^{i-1} N(\mathbf{X}_t | \mathbf{U}_m^{i-1}, \Sigma_m^{i-1})}{\sum_{m=1}^M a_m^{i-1} N(\mathbf{X}_t | \mathbf{U}_m^{i-1}, \Sigma_m^{i-1})}. \end{aligned} \quad (5)$$

Due to the different types of emotions among the datasets, we unify the emotional datasets by categorizing them into positive and negative regions in the valence and arousal dimensions, as shown in Figure 13. We may verify the ability of the emotion classifier by classifying the emotional utterances into different regions in the valence and arousal space.

**4.2. Online Learning Using AdaBoost.** While the offline GMM classifier is trained using EM algorithm, the online training algorithm using AdabBoost will be introduced in this section. AdaBoost is a powerful algorithm in assemble learning [16]. The belief in this AdaBoost is that weak classifiers may be combined into a powerful classifier. Multiple classifiers trained on randomly selected datasets perform quiet differently from each other on the same testing dataset; therefore,

we may reduce the misclassification rate by a proper decision fusion rule.

AdaBoost algorithm consists of several iterations. In each iteration, a new training set is selected for a new weak classifier. A weight is assigned to the new weak classifier. Based on the testing results of the new weak classifier, the weights of all the data samples are modified for the next iteration. At the final step the assembled classifier is achieved by combination of the multiple weak classified through a weighted voting rule.

Let us suppose the current training set is [17]

$$T = \{s_1, s_2, \dots, s_N\}, \quad (6)$$

where the weights of the samples are

$$W = \{w_1, w_2, \dots, w_N\}, \quad (7)$$

$$\sum_{i=0}^N w_i = 1.$$

The error rate of the new weak classifier is

$$e = \sum_{i:c(s_i) \neq y_i} w_i, \quad (8)$$

where  $c(s_i)$  is the classification result and  $y_i$  is the class label. The fusion weight assigned to each classifier is defined by the error rate:

$$\alpha = \ln \left( \frac{1-e}{e} \right). \quad (9)$$

At the beginning of the algorithm, each sample is assigned by equal weight. During the iteration, the sample weights are updated:

$$w_{i+1} = \begin{cases} w_i \times \beta, & c(s_i) \neq y_i, \\ w_i, & c(s_i) = y_i. \end{cases} \quad (10)$$

At the arrival of the new data, assuming that we know the label information for each sample, pretrained classifiers from the offline data are used as initial weak classifiers. AdaBoost algorithm is applied to the new online data, and fusion weights are reassigned to the offline trained classifiers.

At the first  $m$  initial iterations,  $m$  pretrained classifiers are used as the weak classifiers and added to the final ensemble classifier, instead of training new weak classifiers from the randomly selected dataset. After the  $m$  initial iterations, new weak classifiers are trained from the new online data and added to the final ensemble classifier in the AdaBoost algorithm.

The major difference between the online training and the offline training is the data used for learning. Offline training uses large acted data, while online training uses small and natural data. Offline training is independent of the online training and ready to use, while the online training is dependent on the offline training and only retrains the existing model to fit specific purposes, such as to tune on a large number of speakers. The purpose of online training is to quickly adapt the existing offline model to a small amount of new data.

## 5. Experimental Results

In our experiment, the offline training is carried out on the acted basic emotion dataset. The speaker-independent dataset and the elicited practical emotion dataset are used for the online training and the online testing. Although the datasets used in online testing are preprocessed utterances rather than real time online data, our experiments still provide a simulated online situation. We divide dataset 2 and dataset 3 into smaller sets, dataset 2a and dataset 2b, which are used as the simulated online initialization.

Speech utterances from different sources are organized into several datasets, as shown in Table 2.

The online learning algorithm is verified both on the speaker-independent data and the elicited data. The results are shown in Table 4. A large number of speakers bring difficulties in modeling emotional behavior, since emotion expression is highly dependent on individual habit and personality. By extending the offline trained classifier to the online data that contains a large number of speakers, we improved the generality of our SER system. The elicited data is collected in a cognitive experiment that is more close to the real world situation. During the cognitive task emotional speech is induced. We observed that the different nature between the acted data and the induced speech during a cognitive task caused a significant decrease of the recognition rate. By using the online training technique we may transfer the offline trained SER system to the elicited data. Extending our SER system to different data sources may bring emotion recognition closer to real world applications.

The major challenge in our online learning algorithm is how to combine the existing offline classifier and efficiently adapt the model parameters to a small number of new online data. We adopted the incremental learning idea and solved this problem by modifying the initial stage in the AdaBoost framework. One of the contributions of our online learning algorithm is that we may reuse the existing offline training data and make the online learning stage more efficiently. We make use of a large amount of available offline training data and only require a small amount of data for online training, as shown in Table 3. The weight of each weak classifier is an important parameter. The proposed method may be further improved by using fuzzy membership function to evaluate the confidence in GMM classifiers and reestimate the weight of each weak classifier.

## 6. Discussions

Acted data is often considered not suitable for real world applications. However, traditional researches have been focused on the acted emotion speech, and many acted databases are available. How to transfer an SER system that trained on the acted data to the new naturalistic data in real world is an unsolved challenge.

Many feature selection algorithms may be applied to SER system. MIC is a newly proposed and powerful algorithm for exploring nonlinear relationship between variables.

AdaBoost is a popular algorithm to ensemble multiple weak classifiers to establish a strong classifier. By applying

TABLE 3: Selected datasets for online and offline experiments.

Datasets index	Data source	Number of utterances	Purpose of use
Dataset 1	Acted speech	12000	Offline training
Dataset 2a	Speaker independent	1000	Online training
Dataset 2b	Speaker independent	10000	Testing
Dataset 3a	Elicited speech	1000	Online training
Dataset 3b	Elicited speech	5000	Testing

TABLE 4: Online and offline experimental results.

Experiment index	Offline training set	Online training set	Testing set	Classification result %
Experiment 1	Dataset 1	N/A	Dataset 2b	63.3%
Experiment 2	Dataset 1	Dataset 2a	Dataset 2b	75.6%
Experiment 5	Dataset 2a	N/A	Dataset 2b	70.0%
Experiment 3	Dataset 1	N/A	Dataset 3b	61.2%
Experiment 4	Dataset 1	Dataset 3a	Dataset 3b	73.1%
Experiment 6	Dataset 3a	N/A	Dataset 3b	68.5%

AdaBoost in the online occasion, we train multiple weak classifiers based on the newly arrived online data. The offline pretrained classifiers are used for initialization. We may explore other incremental learning algorithms in the future work.

## Acknowledgments

This work was partially supported by China Postdoctoral Science Foundation (no. 2012M520973), National Nature Science Foundation (no. 61231002, no. 61273266, no. 51075068), and Doctoral Fund of Ministry of Education of China (no. 20110092130004). The authors would like to thank the anonymous reviewers for their valuable comments and helpful suggestions.

## References

- [1] C. Clavel, I. Vasilescu, L. Devillers, G. Richard, and T. Ehrette, "Fear-type emotion recognition for future audio-based surveillance systems," *Speech Communication*, vol. 50, no. 6, pp. 487–503, 2008.
- [2] C. Huang, Y. Jin, Y. Zhao, Y. Yu, and L. Zhao, "Speech emotion recognition based on re-composition of two-class classifiers," in *Proceedings of the 3rd International Conference on Affective Computing and Intelligent Interaction and Workshops (ACII '09)*, Amsterdam, The Netherlands, September 2009.
- [3] K. R. Scherer, "Vocal communication of emotion: a review of research paradigms," *Speech Communication*, vol. 40, no. 1-2, pp. 227–256, 2003.
- [4] A. Tawari and M. M. Trivedi, "Speech emotion analysis: exploring the role of context," *IEEE Transactions on Multimedia*, vol. 12, no. 6, pp. 502–509, 2010.
- [5] F. Burkhardt, A. Paeschke, M. Rolfes, W. Sendlmeier, and B. Weiss, "A database of German emotional speech," in *Proceedings*

- of the 9th European Conference on Speech Communication and Technology, pp. 1517–1520, Lissabon, Portugal, September 2005.
- [6] D. Ververidis and C. Kotropoulos, “Automatic speech classification to five emotional states based on gender information,” in *Proceedings of the 12th European Signal Processing Conference*, pp. 341–344, Vienna, Austria, 2004.
  - [7] S. Steidl, *Automatic Classification of Emotion-Related User States in Spontaneous Children’s Speech*, Department of Computer Science, Friedrich-Alexander-Universitaet Erlangen-Nuermberg, Berlin, Germany, 2008.
  - [8] M. Grimm, K. Kroschel, and S. Narayanan, “The Vera am Mittag German audio-visual emotional speech database,” in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME ’08)*, pp. 865–868, Hannover, Germany, June 2008.
  - [9] K. P. Truong, *How Does Real Affect Affect Affect Recognition in Speech?* Center for Telematics and Information Technology, University of Twente, Enschede, The Netherlands, 2009.
  - [10] C. Huang, Y. Jin, Y. Zhao, Y. Yu, and L. Zhao, “Recognition of practical emotion from elicited speech,” in *Proceedings of the 1st International Conference on Information Science and Engineering (ICISE ’09)*, pp. 639–642, Nanjing, China, December 2009.
  - [11] R. Polikar, L. Udpa, S. S. Udpa, and V. Honavar, “Learn++: an incremental learning algorithm for supervised neural networks,” *IEEE Transactions on Systems, Man and Cybernetics C*, vol. 31, no. 4, pp. 497–508, 2001.
  - [12] Q. L. Zhao, Y. H. Jiang, and M. Xu, “Incremental learning by heterogeneous Bagging ensemble,” *Lecture Notes in Computer Science*, vol. 6441, no. 2, pp. 1–12, 2010.
  - [13] R. Xiao, J. Wang, and F. Zhang, “An approach to incremental SVM learning algorithm,” in *Proceedings of the IEEE International Conference on Tools with Artificial Intelligence*, pp. 268–273, 2000.
  - [14] D. N. Reshef, Y. A. Reshef, H. K. Finucane et al., “Detecting novel associations in large data sets,” *Science*, vol. 334, no. 6062, pp. 1518–1524, 2011.
  - [15] D. A. Reynolds and R. C. Rose, “Robust text-independent speaker identification using Gaussian mixture speaker models,” *IEEE Transactions on Speech and Audio Processing*, vol. 3, no. 1, pp. 72–83, 1995.
  - [16] Y. Freund and R. E. Schapire, “A decision-theoretic generalization of on-line learning and an application to boosting,” *Journal of Computer and System Sciences*, vol. 55, no. 1, part 2, pp. 119–139, 1997.
  - [17] Q. Zhao, *The research on ensemble pruning and its application in on-line machine learning [Ph.D. thesis]*, National University of Defense Technology, Changsha, China, 2010.

## Research Article

# A Wavelet-Based Robust Relevance Vector Machine Based on Sensor Data Scheduling Control for Modeling Mine Gas Gushing Forecasting on Virtual Environment

Wang Ting,<sup>1</sup> Cai Lin-qin,<sup>1</sup> Fu Yao,<sup>2</sup> and Zhu Tingcheng<sup>2</sup>

<sup>1</sup> Key Laboratory of Industrial Internet of Things & Networked Control, Ministry of Education, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

<sup>2</sup> Institute of Grassland Science, Key Laboratory of Vegetation Ecology, Ministry of Education, Northeast Normal University, Changchun 130024, China

Correspondence should be addressed to Wang Ting; [wwangting2013@163.com](mailto:wwangting2013@163.com)

Received 20 March 2013; Revised 24 April 2013; Accepted 9 May 2013

Academic Editor: Vishal Bhatnagar

Copyright © 2013 Wang Ting et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

It is wellknown that mine gas gushing forecasting is very significant to ensure the safety of mining. A wavelet-based robust relevance vector machine based on sensor data scheduling control for modeling mine gas gushing forecasting is presented in the paper. Morlet wavelet function can be used as the kernel function of robust relevance vector machine. Mean percentage error has been used to measure the performance of the proposed method in this study. As the mean prediction error of mine gas gushing of the WRRVM model is less than 1.5%, and the mean prediction error of mine gas gushing of the RVM model is more than 2.5%, it can be seen that the prediction accuracy for mine gas gushing of the WRRVM model is better than that of the RVM model.

## 1. Introduction

It is wellknown that mine gas gushing forecasting is very significant to ensure the safety of mining [1–3]. The study results elaborate that the influencing factors of mine gas gushing mainly include buried depth of coal seam, thickness of coal seam, gas content of coal seam, distance of coal seam, daily advance, and day output [4–6]. Artificial neural networks [7–9] have been applied to mine gas gushing forecasting. However, the forecasting results of artificial neural networks are poor due to their shortcomings including over-fitting and local maximum.

A wavelet-based robust relevance vector machine based on sensor data scheduling control for modeling mine gas gushing forecasting is presented in this paper. Relevance vector machine is a kind of improved support vector machine [10–14], which is based on a Bayesian framework. Morlet

wavelet function can be used as the kernel function of robust relevance vector machine.

In the study, the experimental data are employed to study the prediction ability for mine gas gushing. The five factors including buried depth of coal seam, thickness of coal seam, lithology of roof, thickness of bed rock, and ratio of sand and rock are used as the input features of the WRRVM model, and mine gas gushing is used as the output of the WRRVM model. Before the training samples are created, the experimental data must be normalized. Then, the training samples are created. Mean percentage error has been used to measure the performance of the proposed method in this study. As the mean prediction error of mine gas gushing of the WRRVM model is less than 1.5%, and the mean prediction error of mine gas gushing of the RVM model is more than 2.5%, it can be seen that the prediction accuracy for mine gas gushing of the WRRVM model is better than that of the RVM model.

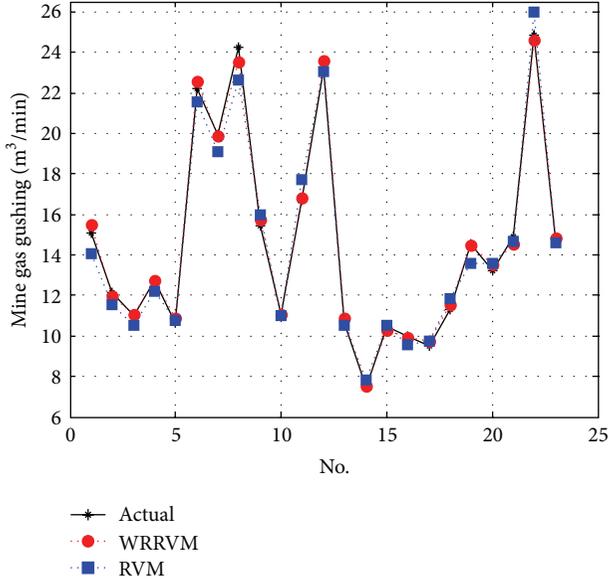


FIGURE 1: The comparison of the prediction values of mine gas gushing between the WRRVM model and the RVM model trained by the five input features.

Therefore, the WRRVM model is very suitable for mine gas gushing prediction.

## 2. Wavelet-Based Robust Relevance Vector Machine

The regression function of relevance vector machine can be described as follows:

$$f(x) = \sum_{i=1}^n w_i \phi(x_i), \quad (1)$$

where  $w_i$  is the weights and  $\phi(\cdot)$  is the mapping function.

Given the vector  $R = [r_1, r_2, \dots, r_n]^T$ , the likelihood function of the dataset can be obtained as follows:

$$p(t | w, r, \sigma^2) = \left( \frac{1}{\sqrt{2\pi\sigma^2}} \right)^n \sqrt{|P|} \times \exp \left\{ -\frac{1}{2\sigma^2} (t - \eta w)' P (t - \eta w) \right\},$$

$$p(w | t, \alpha, r, \sigma^2) = \frac{p(w | \alpha) p(t | w, r, \sigma^2)}{p(t | \alpha, r, \sigma^2)} = N(w | \lambda, \Delta), \quad (2)$$

where  $P = \text{diag}(r_1, r_2, \dots, r_n)$ ,  $\Delta$  is the variance matrix, and  $\lambda$  is the mean value vector, which can be described as follows:

$$\Delta = \left( A + \sigma^{-2} \sum_{i=1}^n r_i \phi(x_i) \phi(x_i)' \right)^{-1},$$

$$\lambda = \sigma^{-2} \Delta \left( \sum_{i=1}^n r_i \phi(x_i) t_i \right). \quad (3)$$

Then, the marginal likelihood function of robust relevance vector machine can be obtained as follows:

$$p(t | \alpha, r, \sigma^2) = \int p(t | w, r, \sigma^2) p(w | \alpha) dw$$

$$= \frac{(1/\sqrt{2\pi})^n}{\sqrt{|\sigma^2/B + \eta A^{-1} \eta'|}} \exp \left\{ -\frac{1}{2} t' \frac{1}{|\sigma^2/B + \eta A^{-1} \eta'|} t \right\}. \quad (4)$$

In this study, Morlet wavelet function can be used as the kernel function of robust relevance vector machine, which can be described as follows [15, 16]:

$$k(x, x') = \prod_{i=1}^m \cos \left( 1.75 \times \frac{x - x'}{a_i} \right) \exp \left( -\frac{\|x - x'\|^2}{2a_i^2} \right). \quad (5)$$

It is sufficient to prove the inequality:

$$F[x](w) = (2\pi)^{-m/2} \int \exp(-j(w \cdot x)) k(x) dx \geq 0, \quad (6)$$

where

$$k(x) = \prod_{i=1}^m \cos \left( \frac{1.75x}{a_i} \right) \exp \left( -\frac{\|x\|^2}{2a_i^2} \right). \quad (7)$$

## 3. Experimental Analysis for Mine Gas Gushing Forecasting Based on Wavelet-Based Robust Relevance Vector Machine

In this section, the experimental data are employed to study the prediction ability for mine gas gushing of the proposed wavelet-based robust relevance vector machine. The five factors including buried depth of coal seam, thickness of coal seam, lithology of roof, thickness of bed rock, and ratio of sand and rock are used as the input features of the WRRVM model, and mine gas gushing is used as the output of the WRRVM model.

Figure 1 gives the comparison of the prediction values of mine gas gushing between the WRRVM model and the RVM model trained by the five input features including buried depth of coal seam, thickness of coal seam, lithology of roof, thickness of bed rock, and ratio of sand and rock; Figure 2 gives the comparison of the prediction values of mine gas gushing between the WRRVM model and the RVM model trained by the four input features including buried depth of coal seam, thickness of coal seam, lithology of roof, and thickness of bed rock; Figure 3 gives the comparison of the prediction values of mine gas gushing between the WRRVM model and the RVM model trained by the four input features including buried depth of coal seam, thickness of coal seam, lithology of roof, and ratio of sand and rock; Figure 4 gives the comparison of the prediction values of mine gas gushing between the WRRVM model and the RVM model trained by

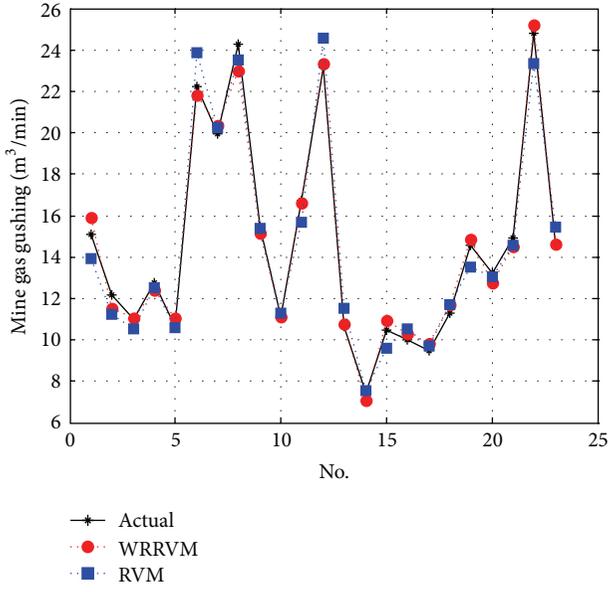


FIGURE 2: The comparison of the prediction values of mine gas gushing between the WRRVM model and the RVM model trained by the four input features including buried depth of coal seam, thickness of coal seam, lithology of roof, and thickness of bed rock.

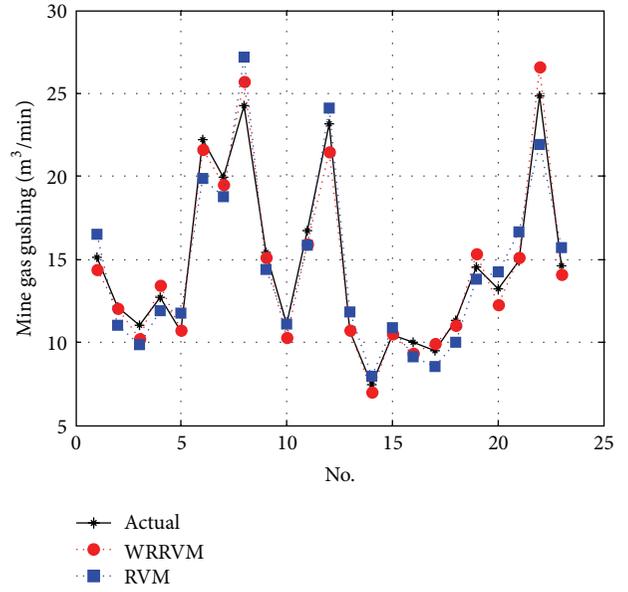


FIGURE 4: The comparison of the prediction values of mine gas gushing between the WRRVM model and the RVM model trained by the three input features including buried depth of coal seam, thickness of coal seam, and lithology of roof.

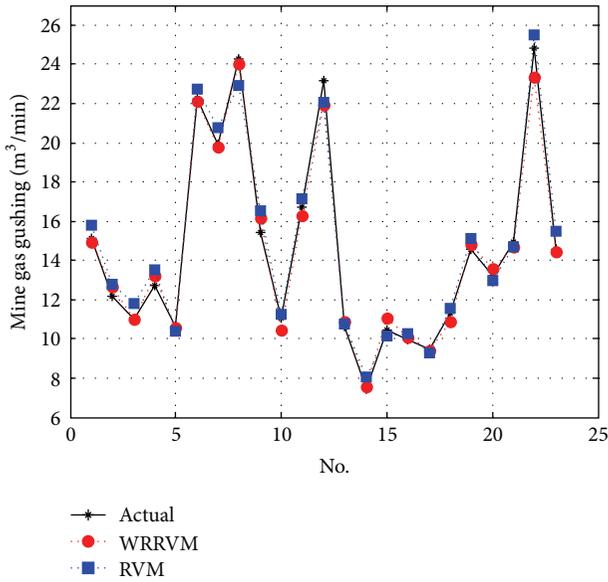


FIGURE 3: The comparison of the prediction values of mine gas gushing between the WRRVM model and the RVM model trained by the four input features including buried depth of coal seam, thickness of coal seam, lithology of roof, and ratio of sand and rock.

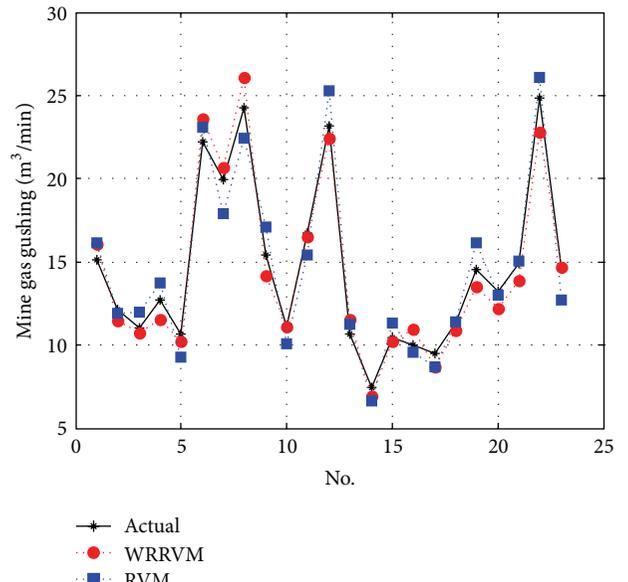


FIGURE 5: The comparison of the prediction values of mine gas gushing between the WRRVM model and the RVM model trained by the three input features including buried depth of coal seam, thickness of coal seam, and ratio of sand and rock.

the three input features including buried depth of coal seam, thickness of coal seam, and lithology of roof; Figure 5 gives the comparison of the prediction values of mine gas gushing between the WRRVM model and the RVM model trained by the three input features including buried depth of coal seam, thickness of coal seam, and ratio of sand and rock. Mean percentage error has been used to measure the performance

of the proposed method in this paper. As the mean percentage prediction error of mine gas gushing of the WRRVM model is less than 1.5%, and the mean percentage prediction error of mine gas gushing of the RVM model is more than 2.5%, it can be seen that the prediction accuracy for mine gas gushing of the WRRVM model is better than that of the RVM model.

#### 4. Conclusion

A wavelet-based robust relevance vector machine based on sensor data scheduling control for modeling mine gas gushing forecasting is presented in the paper. Morlet wavelet function can be used as the kernel function of robust relevance vector machine. As the mean prediction error of mine gas gushing of the WRRVM model is less than 1.5%, and the mean prediction error of mine gas gushing of the RVM model is more than 2.5%, it can be seen that the prediction accuracy for mine gas gushing of the WRRVM model is better than that of the RVM model. Therefore, the WRRVM model is very suitable for mine gas gushing prediction.

#### Acknowledgments

The work is supported by the National Natural Science Foundation Project of NSFC (NSFC, 50804061), the Chongqing Education Administration Program Foundation of China (no. KJ120514), Chongqing Education Administration Program Foundation of China (no. KJ110513), and the Natural Science Foundation Project of CQ CSTC (no. 2012BB3725). This work was supported in part by the Foundation for University Youth Key Teacher of Chongqing, Outstanding Achievement Transformation Project of CQ CQJW (no. KJzh10207). And special thanks are due to all who have helped to make this study.

#### References

- [1] J. Zhang, "The application of analytic hierarchy process in mine gas prevention system," pp. 1576–1584.
- [2] R. Packham, Y. Cinar, and R. Moreby, "Simulation of an enhanced gas recovery field trial for coal mine gas management," *International Journal of Coal Geology*, vol. 85, no. 3-4, pp. 247–256, 2011.
- [3] W. Yang, B. Lin, C. Zhai, X. Li, and S. An, "How in situ stresses and the driving cycle footage affect the gas outburst risk of driving coal mine roadway," *Tunnelling and Underground Space Technology*, vol. 31, pp. 139–148, 2012.
- [4] K. X. Wang, X. H. Fu, Y. A. Zhou, Y. HE, and H. Wu, "Dynamic development characteristics of amounts of gas and levels of pressure in the Pan-1 coal mine of Huainan," *Mining Science and Technology*, vol. 19, no. 6, pp. 740–744, 2009.
- [5] W. Wang and J. Yan, "The geological factor analysis of influenced Tianchi coal mine gas occurrence," *Procedia Engineering*, vol. 45, pp. 317–321, 2012.
- [6] L. Wang, Y. P. Cheng, L. Wang, P. K. Guo, and W. Li, "Safety line method for the prediction of deep coal-seam gas pressure and its application in coal mines," *Safety Science*, vol. 50, no. 3, pp. 523–529, 2012.
- [7] B. Lamrini, G. Della Valle, I. C. Trelea, N. Perrot, and G. Trystram, "A new method for dynamic modelling of bread dough kneading based on artificial neural network," *Food Control*, vol. 26, no. 2, pp. 512–524, 2012.
- [8] E. K. Lafdani, A. M. Nia, and A. Ahmadi, "Daily suspended sediment load prediction using artificial neural networks and support vector machines," *Journal of Hydrology*, vol. 478, pp. 50–62, 2013.
- [9] M. Khashei and M. Bijari, "A novel hybridization of artificial neural networks and ARIMA models for time series forecasting," *Applied Soft Computing Journal*, vol. 11, no. 2, pp. 2664–2675, 2011.
- [10] C. C. Chuang and Z. J. Lee, "Hybrid robust support vector machines for regression with outliers," *Applied Soft Computing Journal*, vol. 11, no. 1, pp. 64–72, 2011.
- [11] J. Park, I. H. Kwon, S. S. Kim, and J. G. Baek, "Spline regression based feature extraction for semiconductor process fault detection using support vector machine," *Expert Systems with Applications*, vol. 38, no. 5, pp. 5711–5718, 2011.
- [12] A. A. Ensafi, F. Hasanpour, T. Khayamian, A. Mokhtari, and M. Taei, "Simultaneous chemiluminescence determination of thebaine and noscapine using support vector machine regression," *Spectrochimica Acta A*, vol. 75, no. 2, pp. 867–871, 2010.
- [13] B. Apolloni, D. Malchiodi, and L. Valerio, "Relevance regression learning with support vector machines," *Nonlinear Analysis, Theory, Methods and Applications*, vol. 73, no. 9, pp. 2855–2864, 2010.
- [14] E. Fokoué, D. Sun, and P. Goel, "Fully Bayesian analysis of the relevance vector machine with an extended hierarchical prior structure," *Statistical Methodology*, vol. 8, no. 1, pp. 83–96, 2011.
- [15] W. Liu and B. Tang, "A hybrid time-frequency method based on improved Morlet wavelet and auto terms window," *Expert Systems with Applications*, vol. 38, no. 6, pp. 7575–7581, 2011.
- [16] B. Tang, W. Liu, and T. Song, "Wind turbine fault diagnosis based on Morlet wavelet transformation and Wigner-Ville distribution," *Renewable Energy*, vol. 35, no. 12, pp. 2862–2866, 2010.

## Research Article

# Neural Model with Particle Swarm Optimization Kalman Learning for Forecasting in Smart Grids

Alma Y. Alanis,<sup>1</sup> Luis J. Ricalde,<sup>2</sup> Chiara Simetti,<sup>3</sup> and Francesca Odone<sup>3</sup>

<sup>1</sup> CUCEI, Universidad de Guadalajara, Apartado Postal 51-71, Colonia Las Aguilas, 45080 Zapopan, JAL, Mexico

<sup>2</sup> UADY, Faculty of Engineering, Avenida Industrias no Contaminantes por Periferico Norte, Apartado Postal 115 Cordemex, Merida, Yuc, Mexico

<sup>3</sup> DISI, Università degli Studi di Genova, Via Dodecaneso 35, 16146 Genova, Italy

Correspondence should be addressed to Alma Y. Alanis; [almayalanis@gmail.com](mailto:almayalanis@gmail.com)

Received 29 March 2013; Accepted 27 May 2013

Academic Editor: Yudong Zhang

Copyright © 2013 Alma Y. Alanis et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper discusses a novel training algorithm for a neural network architecture applied to time series prediction with smart grids applications. The proposed training algorithm is based on an extended Kalman filter (EKF) improved using particle swarm optimization (PSO) to compute the design parameters. The EKF-PSO-based algorithm is employed to update the synaptic weights of the neural network. The size of the regression vector is determined by means of the Cao methodology. The proposed structure captures more efficiently the complex nature of the wind speed, energy generation, and electrical load demand time series that are constantly monitored in a smart grid benchmark. The proposed model is trained and tested using real data values in order to show the applicability of the proposed scheme.

## 1. Introduction

The limited existing reserves of fossil fuels and the harmful emissions associated with them have led to an increased focus on renewable energy applications in recent years. The first steps on integrating renewable energy sources began with hybrid wind and solar systems as complementing sources and as a solution for rural applications and weak grid interconnections. Further research has implemented hybrid systems including several small scale renewable energy sources as solar thermal, biomass, fuel cells, and tidal power. Since the production costs for photovoltaic and wind turbine applications have considerably reduced, they have become the primary choice for hybrid energy generation systems. The future of energy production is headed towards this scheme of integration of renewable energy sources with existing conventional generation systems with a high degree of measurement, communications, and control. This integration is defined as a smart grid. This new scheme increases the power quality since the production becomes decentralized and is the main reason for which institutions have increased the research on this concept [1]. Microgrids integrate small scale

energy generation systems mainly from renewable energy and implement complex control technologies to improve the flexibility and reliability of the power system. The design of these systems integrates a distributed power generation system and a management unit composed of a communication network which monitor and controls the interconnection between energy sources, storage devices, and electrical loads.

Among renewable energy sources, wind energy is the one with the lowest cost of electricity production [2]. However, in practice the integration of wind energy into the existing electricity supply system is a real challenge because its availability mainly depends on meteorological conditions, particularly on the magnitude of the wind speed, which cannot directly be changed by human intervention. For this reason, it is important to have a reliable estimation of wind velocity and direction which directly affects the energy generation. Integration of the forecast of wind speed and output power is a good way to improve the performance in scheduling for smart grids [3]. Wind prediction is not an easy task; the wind has a stochastic nature with high rate of change. Wind speed time series present highly nonlinear behavior with no typical patterns and a weak seasonal character [4].

Several methods have been proposed to accomplish wind characteristics forecasting like numerical weather prediction systems, statistical approaches, and artificial neural networks using feedforward or recurrent structures [2, 5–9]. In [6], a linear-time-series-based model relating the predicted interval to its corresponding one and data covering a temporal span of two years is developed. The statistical approaches have the advantage of low cost since they only require historical data; on the other hand, the accuracy of the prediction drops for long time horizons. Artificial intelligence methods are more suitable for short-term predictions; these methods are based on time series historical data in order to build a mathematical model which approximates the input-output relationship. Time-series-based models include the autoregressive (AR) and the autoregressive moving average (ARMA) models which predict the average wind speed for one step ahead [9].

For wind generation systems, artificial neural networks (ANN) have been considered as a convenient analysis tool for wind energy systems forecasting and control applications due to the simplicity of the model and the accuracy of the results for nonlinear and stochastic models and have been implemented in several practical applications [8]. In [10], a merged neuro-fuzzy system is developed as a universal approximator in order to estimate the state of charge in a battery bank. For wind generation systems, ANN have been considered as a convenient analysis tool for wind forecasting due to the simplicity of the model and the accuracy of the results.

In [8], a recurrent neural network is applied to forecast the output power of wind generators based on wind speed prediction using one year of historical data achieving from hour-ahead to day-ahead predictions with errors ranging from 5% for one hour horizon to 20% for one day ahead forecasting. In [5], local recurrent neural networks are implemented to forecast wind speed and electrical power in a wind park with a seventy-two hour ahead forecast and obtaining a better performance in comparison with static network approaches.

ANN have been previously implemented for wind power short-term predictions, outperforming other classical methods due to the fast learning algorithm which enables on-line implementations and the versatility to vary the prediction horizon [7]. Due to their nonlinear modeling characteristics, neural networks have been successfully applied in control systems, pattern classification, pattern recognition, and time series forecasting problems. There are several previous works that use artificial neural networks to predict wind time series [2, 4, 11]. The best well-known training approach for recurrent neural networks (RNN) is the back propagation through time [12]. However, it is a first-order gradient descent method, and hence its learning speed could be very slow [13]. Another well-known training algorithm is the Levenberg-Marquardt one [14]; its principal disadvantage is that it is not guaranteed that it will find the global minimum and its learning speed could be slow too, and this depends on the initialization. In past years, extended-kalman-filter- (EKF-) based algorithms have been introduced to train neural networks [15, 16]. With the EKF-based algorithm, the learning convergence is improved [13]. The EKF training of neural networks, both

feedforward and recurrent ones, has proven to be reliable for many applications over the past ten years [16]. However, EKF training requires the heuristic selection of some design parameters which is not always an easy task [11, 15].

During the past decade, the use of evolutionary computation in engineering applications has increased. Evolutionary algorithms apply adaptation and stochastically in optimization problems in schemes such as evolutionary programming, genetic algorithms, and evolution strategies [17]. Particle swarm optimization (PSO) technique, which is based on the behavior of a flock of birds or school of fish, is a type of evolutionary computing technique [18]. The PSO algorithm uses a population of search points that evolve in a search space using a communication method to transfer the acquired experience from best solutions. This algorithm has several advantages like the simplicity of the updating law, faster convergence time, and less complexity on the reorganization of the population. The PSO methods also have emerged as an excellent tool to improve the performance of neural network learning process [19]. In [20], a PSO learning rule for a neural network is implemented using FPGA for dynamic system identification. In [21], the PSO algorithm is extended to multiple swarms in a neuro-fuzzy network with good results in forecasting applications. It has been shown that the PSO training algorithm takes fewer computations and is faster than the BP algorithm for neural networks to achieve the same performance [18].

In this paper, we propose the use of PSO for tuning the parameters of EKF training algorithm. This scheme is a new one and is suitable for data modeling in smart grids since the forecasting horizon satisfies the requirements for several applications in the grid operation. The length of the regression vector is determined using the Cao methodology which is an improvement to the false neighbors approach [22]. The applicability of this architecture is illustrated via simulation using real data values for electric load demand (ELD), wind speed (WS), and wind energy generation (WEG) in order to show the potential applications in forecasting for energy generation in smart grid schemes.

The remainder of the paper is organized as follows. Section 2 is devoted to describing the neural model, based on the recurrent multilayer perceptron (RMLP), where the training phase relies on an extended Kalman filter which is able to deal with the nonlinearity of the model, and the initialization of the system is based on a particle swarm optimization strategy. Section 3 reports the experimental analysis of the proposed method applied to the problem of predicting variables in smart grids. Finally Section 4 includes the conclusions and future work.

## 2. Neural Identification

In this paper for the neural model identification, the recurrent multilayer perceptron is chosen, and then the neural model structure problem reduces to dealing with the following issues: (1) selecting the inputs to the network and (2) selecting the internal architecture of the network.

The structure selected in this paper is NNARX [14] (acronym for neural network autoregressive external input);

the output vector for the artificial neural network is defined as the regression vector of an autoregressive external input linear model structure (ARX) [15].

It is common to consider a general nonlinear system; however, for many control applications it is preferred to express the model in an affine form, which can be represented by the following equations:

$$y(k+1) = f(y(k), y(k-1), \dots, y(k-q+1)), \quad (1)$$

where  $q$  is the dimension of the regression vector. In other words, a nonlinear mapping  $f$  exists, for which the present value of the output  $y(k+1)$  is uniquely defined in terms of its past values  $y(k), \dots, y(k-q+1)$  and the present values of the input  $u(k)$ .

Considering that it is possible to define

$$\phi(k) = [y(k), \dots, y(k-q+1)]^T \quad (2)$$

which is similar to the regression vector of a ARX linear model structure [14], then the nonlinear mapping  $f$  can be approximated by a neural network defined as

$$y(k+1) = \varphi(\phi(k), w^*) + \varepsilon, \quad (3)$$

where  $w^*$  is an ideal weight vector, and  $\varepsilon$  is the modeling error; such neural network can be implemented on predictor form as

$$\hat{y}(k+1) = \varphi(\phi(k), w), \quad (4)$$

where  $w$  is the vector containing the adjustable parameters in the neural network.

The neural network structure used in this work is depicted in Figure 1, which contains sigmoid units only in the hidden layer; the output layer is a linear one. The used sigmoid function  $S(\cdot)$  is defined as a logistic one as follows:

$$S(\varsigma) = \frac{1}{1 + \exp(-\beta\varsigma)}, \quad \beta > 0, \quad (5)$$

where  $\varsigma$  is any real value variable.

**2.1. EKF Training Algorithm.** Kalman filter (KF) estimates the state of a linear system with additive state and output white noise. Kalman filter algorithm is developed for a linear, discrete-time dynamical system. For KF-based neural network training, the network weights become the states to be estimated. Due to the fact that the neural network mapping is nonlinear, an EKF type is required [15].

Consider a nonlinear dynamic system described by the next model in state space

$$\begin{aligned} w(k+1) &= f(k, w(k)) + v_1(k), \\ y(k) &= h(k, w(k)) + v_2(k), \end{aligned} \quad (6)$$

where  $v_1(k)$  and  $v_2(k)$  are zero-mean, white noises with covariance matrices given by  $Q(k)$  and  $R(k)$ , respectively.  $f(k, w(k))$  denotes the nonlinear transition matrix function.

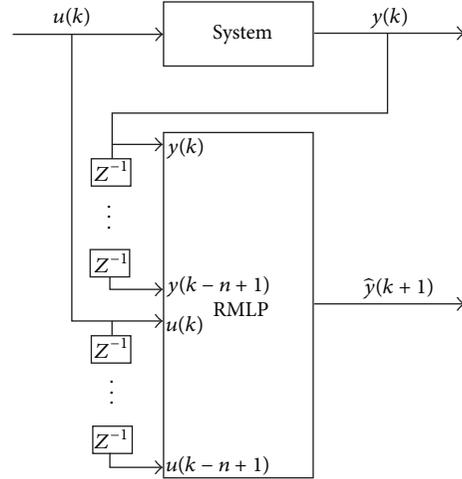


FIGURE 1: Neural network structure.

The basic idea of the extended Kalman filter is to linearize the state space model 4 at each time instant around the most recent state estimate, which is taken to be  $\hat{w}(k)$ . The training goal is to find the optimal weight values which minimize the prediction error. The modified extended kalman filter (EKF) algorithm is defined by

$$\hat{w}(k+1) = \hat{w}(k) + K(k) [y(k) - \hat{y}(k)],$$

$$K(k) = P(k) H^T(k) M(k), \quad (7)$$

$$P(k+1) = P(k) - K(k) H(k) P(k) + Q(k)$$

with

$$M(k) = [R(k) + H(k) P(k) H^T(k)]^{-1}, \quad (8)$$

$$e(k) = y(k) - \hat{y}(k),$$

where  $e(k) \in \mathfrak{R}$  is the respective approximation error,  $P \in \mathfrak{R}^{L \times L}$  is the prediction error associated covariance matrix at step  $k$ ,  $w \in \mathfrak{R}^L$  is the weight (state) vector,  $L$  is the respective number of neural network weights,  $y$  is the system output,  $\hat{y}$  is the neural network output,  $K \in \mathfrak{R}^L$  is the Kalman gain vector,  $Q \in \mathfrak{R}^{L \times L}$  is the state noise associated covariance matrix,  $R \in \mathfrak{R}$  is the measurement noise associated covariance,  $H \in \mathfrak{R}$  is a vector, in which each entry ( $H_{ij}$ ) is the derivative of one of the neural network outputs, ( $\hat{y}$ ), with respect to one neural network weight, and ( $w_j$ ) is defined as follows:

$$H_{ij}(k) = \left[ \frac{\partial \hat{y}(k)}{\partial w_j(k)} \right]_{w_j(k) = \hat{w}_j(k)}, \quad (9)$$

where  $i = 1, \dots, m$ ;  $j = 1, \dots, L$ . Usually  $P$ ,  $Q$ , and  $R$  are initialized as diagonal matrices, with entries  $P(0)$ ,  $Q(0)$ , and  $R(0)$ , respectively. It is important to note that for the EKF training algorithm  $P(0)$ ,  $Q(0)$ , and  $R(0)$  are considered as design parameters that are typically heuristically determined; however, in this paper we propose the use of particle swarm optimization for determining such parameters [15].

*2.2. PSO Improvement for EKF Training Algorithm.* Particle swarm optimization (PSO) is a swarm intelligence technique developed by Kennedy and Eberhart in 1995 [23]. In fact, natural flocking and swarm social behavior of birds and insects inspired the PSO. This technique has been used in several optimization and engineering problems [2, 18, 24]. In the basic PSO technique proposed by Kennedy and Eberhart [23], a great number of particles move around in a multidimensional space and each particle memorizes its position vector and velocity vector as well as the time at which the particle has acquired the best fitness. Furthermore, related particles can share data at the best-fitness time. The velocity of each particle is updated with the best positions acquired for all particles over iterations and the best positions are acquired by the related particles over generations.

To improve the performance of the basic PSO algorithm, some new versions of it have been proposed. At first, the concept of an inertia weight is developed to better control exploration and exploitation in [18, 25]. Then, the research done by Clerc [26] indicated that using a constriction factor may be necessary to insure convergence of the particle swarm algorithm. After these two important modifications in the basic PSO, the multiphase particle swarm optimization (MPSO), the particle swarm optimization with Gaussian mutation, the quantum particle swarm optimization, a modified PSO with increasing inertia weight schedule, the Gaussian particle swarm optimization (GPSO), and the guaranteed convergence PSO (GCP SO) were introduced in [27], respectively.

In this paper, the algorithm proposed in [18] is used in order to determine the design parameters for the EKF-learning algorithm. Initially, a set of random solutions or a set of particles are considered. A random velocity is given to each particle and they are flown through the problem space. Each particle has a memory which is used to keep track of the previous best position and corresponding fitness. The best value of the position of each individual is stored as  $p_{id}$ . In other words,  $p_{id}$  is the best position acquired by an individual particle during the course of its movement within the swarm. It has another value called the  $p_{gd}$ , which is the best value of all the particles  $p_{id}$  in the swarm. The basic concept of the PSO technique lies in accelerating each particle towards its  $p_{id}$  and  $p_{gd}$  locations at each time step. The PSO algorithm used in this paper is defined as follows [18].

- (1) Initialize a population of particles with random positions and velocities in the problem space.
- (2) For each particle, evaluate the desired optimization fitness function.
- (3) Compare the particles fitness evaluation with the particles  $p_{id}$  and if the current value is better than the  $p_{id}$ , then set  $p_{id}$  value equal to the current location.
- (4) Compare the best fitness evaluation with the population's overall previous best. If the current value is better than the  $p_{gd}$ , then set  $p_{gd}$  to the particle's array and index value.
- (5) Update the particle's velocity and position as follows.

The velocity of the  $i$ th particle of  $d$  dimension is given by

$$v_{id}(k+1) = c_0 v_{id}(k) + c_1 \text{rand}_1(p_{id}(k) - x_{id}(k)) + c_2 \text{rand}_2(p_{gd}(k) - x_{id}(k)). \quad (10)$$

The position vector of the  $i$ th particle of  $d$  dimension is updated as follows:

$$x_{id}(k+1) = x_{id}(k) + v_{id}(k), \quad (11)$$

where  $c_0$  is the inertia weight,  $c_1$  is the cognition acceleration constant, and  $c_2$  is the social acceleration constant.

- (6) Repeat step (2) until a criterion is met, usually a sufficiently good fitness or a maximum number of iterations or epochs.

In case the velocity of the particle exceeds  $V_{\max}$  (the maximum velocity for the particles), then it is reduced to  $V_{\max}$ . Thus, the resolution and fitness of search depends on the  $V_{\max}$ . If  $V_{\max}$  is too high, then particles will move in larger steps and so the solution reached may not be as good as expected. If  $V_{\max}$  is too low, then particles will take a long time to reach the desired solution [18]. The above explained PSO are very suitable models of noisy problems, as the one we are considering.

*2.3. Regressor Structure.* We now discuss the choice of an appropriate number of delayed signals to be used in the training phase; a wrong number of delayed signals, used as regressors, could have a substantially negative impact on the training process, while a too small number imply that essential dynamics will not be modeled. Additionally, a large number of regression terms increase the required computation time. Also, if too many delayed signals are included in the regression vector, it will contain redundant information. For a good behavior of the model structure, it is necessary to have both a sufficiently large lag space and an adequate number of hidden units. If the lag space is properly determined, the model structure selection problem is substantially reduced. There have been many discussions of how to determine the optimal embedding dimension from a scalar time series based on Takens' theorem [22]. The basic methods, which are usually used to choose the minimum embedding dimension, are (1) computing some invariants on the attractor, (2) singular value decomposition, and (3) the method of false neighbors. However, a practical method to select the lag space is the one proposed by Cao [22] to determine the minimum embedding dimension; it overcomes most of the shortcomings of the above mentioned methodologies, like high dependence from design parameters and high computational cost, among others [22]. Besides, it has several advantages: it does not contain any subjective parameters except for the time-delay embedding; it does not strongly depend on how many data points are available; it can clearly distinguish deterministic signals from stochastic signals; it works well for time series from high-dimensional attractors, and it is computationally

efficient. In this paper, this technique for determination of the optimal regressor structure is used.

Let us consider a time series  $x_1, x_2, \dots, x_n$  and define a set of time-delay vectors as

$$y_i = [x_i \ x_{i+\tau} \ \dots \ x_{i+(d-1)\tau}], \quad (12)$$

$$i = 1, 2, \dots, N - (d-1)\tau,$$

where  $d$  is the embedding dimension. This dimension is determined from the evolution of a function  $E(d)$  defined as

$$E(d) = \frac{1}{N - d\tau} \sum_{i=1}^{N-d\tau} \frac{\|y_i(d+1) - y_{n(i,d)}(d+1)\|}{\|y_i(d) - y_{n(i,d)}(d)\|}, \quad (13)$$

$$i = 1, 2, \dots, N - d\tau,$$

where  $n(i, d)$  is an integer such that  $y_{n(i,d)}(d)$  is the nearest neighbor of  $y_i(d)$  [28]. The minimum embedding dimension  $d_0 + 1$  is determined when  $E(d)$  stops changing for any  $d_0$ .

### 3. Simulation Results

In this section, two application examples to validate the proposed PSO-EKF learning algorithm are presented. First, the experimental analysis of the proposed method applied to the problem of predicting the wind speed in order to compare the performance with similar approaches [29] is discussed. As a second test for the proposed method, the neural predictor with data obtained for the microgrid in the UADY School of Engineering is implemented in order to evaluate the performance with time series of a different nature but related with the energy production and demand in smart grids.

**3.1. Comparison of the PSO Algorithm for Wind Speed Forecasting.** In order to evaluate the performance of the PSO algorithm and compare with similar methods, a neural network predictor for wind speed is implemented, on the basis of the proposed training algorithm. The proposed algorithm requires the following methodology.

- (1) Define the training set. Training is performed using minute data from the first 3 hours from January 1, 2011 and the testing is performed using the 3 hours subsequent to the data training. Experimental data is taken from [30].
- (2) Determine the optimal dimension of the regression vector (1) for dataset of step (1).
- (3) Select the neural structure to be used (4).
- (4) Train the neural identifier.
- (5) Validate the neural identifier using the testing data.

The neural network used is an RMLP trained with an PSO-EKF, whose structure is presented in Figure 1; the hidden layer has 5 units with logistic sigmoid activation functions (5), whose  $\beta$  is fixed in 1 and the output layer is composed of just one neuron, with a linear activation function.

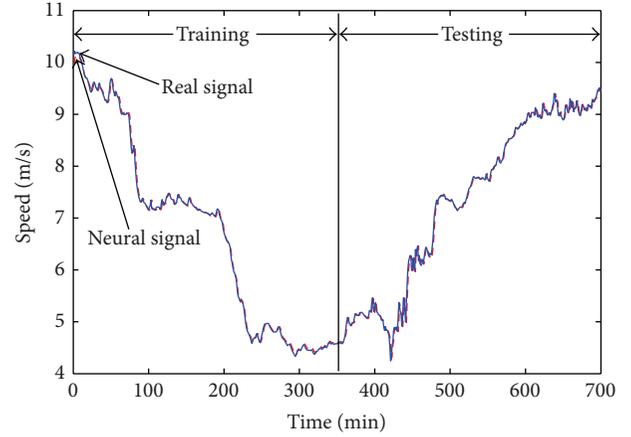


FIGURE 2: Identification performance for wind speed forecast.

TABLE 1: Mean value ( $\mu$ ) and standard deviation ( $\sigma$ ) for identification error.

	PSO-EKF	EKF	LM
$\mu$	$-5.6693 \times 10^{-6}$	$-1.2547 \times 10^{-5}$	$1.8354 \times 10^{-4}$
$\sigma$	0.0749	0.0807	0.0724

The initial values for the covariance matrices ( $R, Q, P$ ) are determined using the PSO algorithm, with 200 as the maximum number of iterations, 4 generations, 3 particles, and  $c_1 = c_2 = 0.1$ . The initial values for neural weights are randomly selected. The length of the regression vector is 5 because that is the order of the system, which is determined using the Cao methodology.

The training is performed offline, using a series-parallel configuration; for this case the delayed output is taken from the wind speed. The mean square error (MSE) reached in training is  $1.735 \times 10^{-5}$  in 200 iterations and the mean absolute relative error reached is 0.6912%. Besides, to measure the performance of the neural network, the absolute relative error (ARE) is calculated from

$$\text{ARE} = \left| \frac{y(n) - \hat{y}(n)}{y(n)} \right|, \quad (14)$$

where  $\hat{y}(n)$  is the predicted wind speed time series achieved by the network. Simulation results are presented as follows: Figure 2 displays the wind speed time series neural identification, Figure 3 includes the identification error (8), Figure 4 shows the time evolution of mean square error, and Figure 5 displays the absolute relative error obtained with (14).

Figure 6 depicts the comparison detail of the proposed PSO-EKF training algorithm against the classical EKF one and the well-known Levenberg-Marquardt one.

Table 1 includes a comparison between the proposed PSO-EKF learning algorithm, the EKF one and the well-known Levenberg-Marquardt (LM) one.

Results included in Table 1 show that the proposed methodology leads to an improvement of the results. Therefore for the second example only PSO-EKF results are presented.

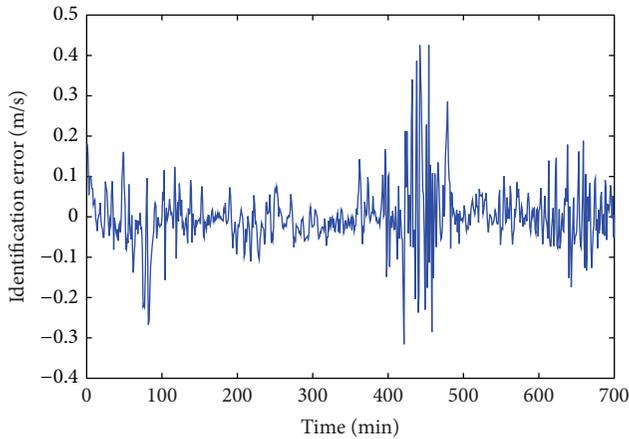


FIGURE 3: Identification error.

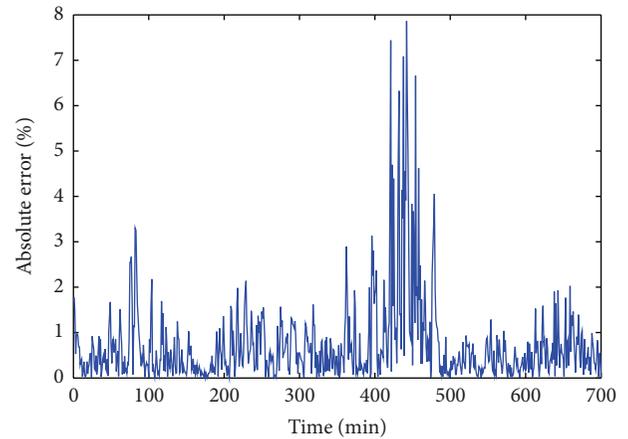


FIGURE 5: Absolute relative error.

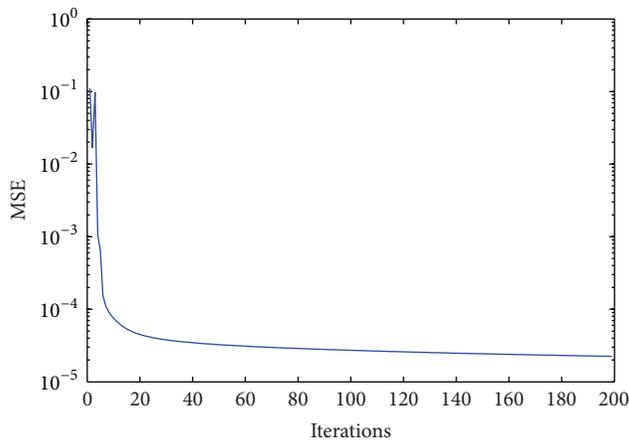


FIGURE 4: Mean square error.

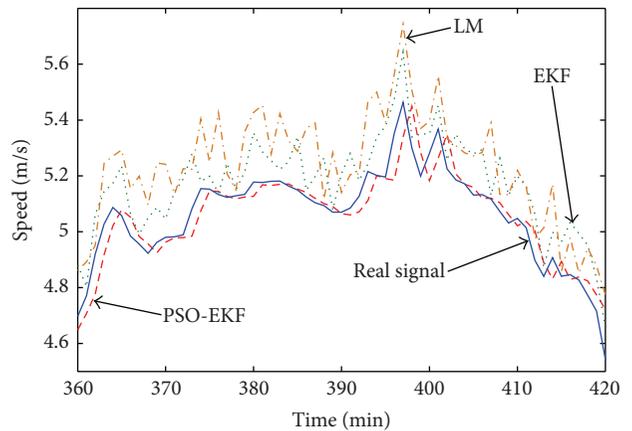


FIGURE 6: Identification performance comparison for wind speed forecast.

**3.2. Forecasting Results for Smart Grid Variables.** Due to random variations in weather conditions, power generation from renewable sources is constantly changing. Combining the forecast of wind speed and output power is a good way to improve the performance in scheduling of wind power [3]. Reliability is one of the most important factors in smart grid operation, so constant monitoring and control is necessary to achieve this goal. An accurate forecast can improve the performance of intelligent controllers and management systems in the grid.

This project is implemented in the Mechatronics Building of the UADY Faculty of Engineering using the data obtained from a 10 kW wind turbine as shown in Figure 7. To characterize the energy consumption from the public grid a year of data has been collected. To characterize the wind and solar potential, irradiance and wind speed data are collected from the meteorological station installed in the FI-UADY. The statistical values obtained from a one-year analysis are applied to train the neural predictor.

Figures 8, 10, and 12 display the computation of the minimum embedding dimension for each one of the analyzed time series. We select 6 regressors to be included in the neural



FIGURE 7: Wind turbines and photovoltaics modules in the FI-UADY.

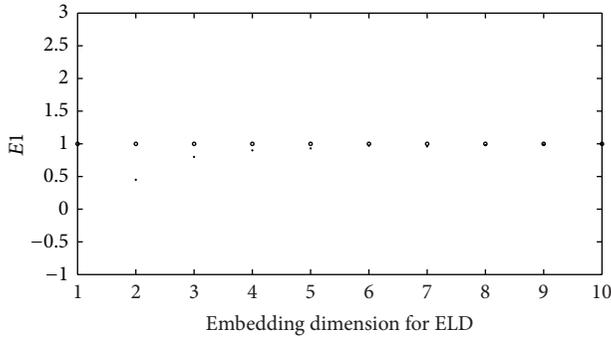


FIGURE 8: Embedding dimension for the electrical load demand time series.

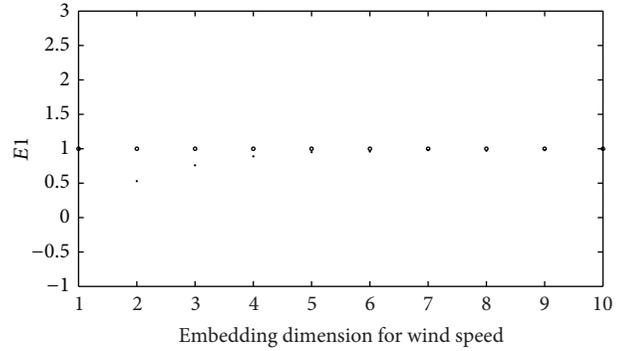


FIGURE 10: Embedding dimension for wind speed.

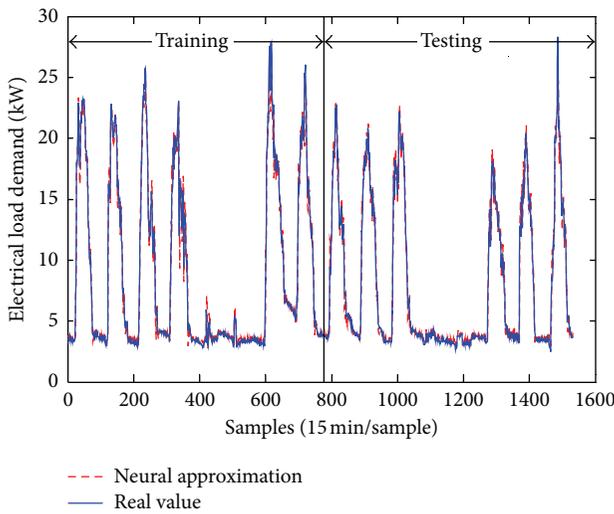


FIGURE 9: Electric load demand time series forecasting.

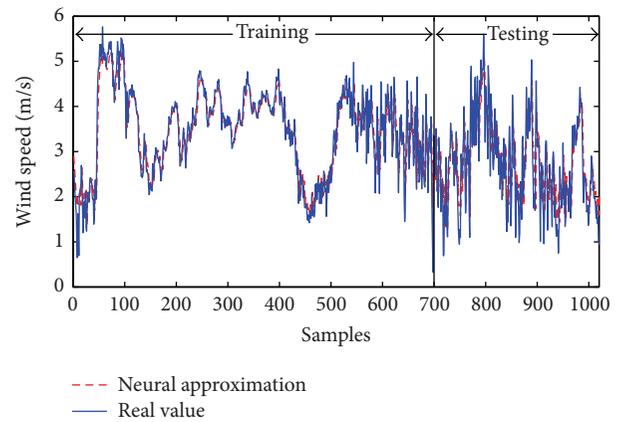


FIGURE 11: Wind speed time series forecasting.

network input vector for the electrical load demand (ELD), 8 for wind energy generation (WEG), and 7 for the wind speed (WS). To train the HONN for each variable, we kept the following design parameters: 2 external inputs corresponding to hours and days, 25 units in the hidden layer, 1 neuron in the output layer, 300 iterations maximum, initial values for synaptic weights randomly selected in the range, and MSE required to end the training less than  $1 \times 10^{-4}$ . The training is performed offline, using a parallel configuration; for this case the delayed output is taken from the neural network output. The initial values for the covariance matrices are for the electrical load demand (ELD) and for the wind speed (WS). The data for the ELD is collected every 5 minutes and averaged each 15 minutes, in the case of the WS which is taken every minute and without average; therefore, each variable is plotted as a function of the sample. The data for the wind energy generation (WEG) is collected every 18 minutes.

For the ELD, 765 are used as samples to accomplish the network training. The data size to train the neural network for WS forecasting is 715 samples. In order to verify if the proposed scheme is adequate using less samples, 200 samples are employed for the WEG and good results are obtained

using this reduced samples number, as exemplified by the simulation results. The results for the ELD, WS, and WEG time series forecasting are shown. As can be seen from Figures 9, 11, and 13, the forecasting is successfully done with a good prediction horizon.

#### 4. Conclusions

This paper proposes the use of an RMLP trained with a PSO-EKF learning algorithm, to predict minutely wind speed with good results as shown in Table 1. The proposed method has a compact structure but taking into account the dynamic nature of the system where behavior is required to be predicted. The proposed neural identifier proved in our experiments to be a model that captures very well the complexity associated with important variables in smart grids operation. Future work on implementing higher order neural networks aims for the design of optimal operation algorithms for smart grids composed of wind and photovoltaic generation systems interconnected to the utility grid. This management system can use the forecasting data to operate the global system, fulfilling the load demand, minimizing the power supplied by the utility grid, and maximizing the one supplied by renewable sources.

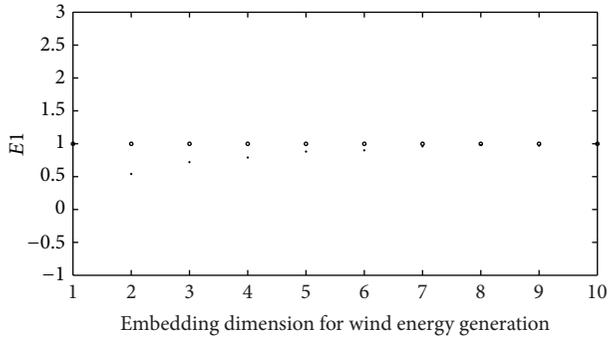


FIGURE 12: Embedding dimension for wind energy generation.

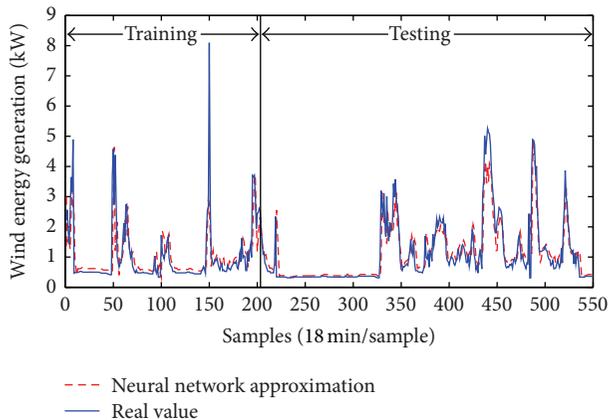


FIGURE 13: Wind energy generation forecasting.

## Acknowledgments

The authors thank the support of CONACYT Mexico, through Project 103191Y and FOMIX 170414. They also thank the very useful comments of the anonymous reviewers, which helped to improve the paper.

## References

- [1] M. Meiqin, D. Ming, S. Jianhui, L. Chang, S. Min, and Z. Guorong, "Testbed for microgrid with multi-energy Generators," in *Proceedings of the IEEE Canadian Conference on Electrical and Computer Engineering (CCECE '08)*, pp. 637–640, May 2008.
- [2] R. L. Welch, S. M. Ruffing, and G. K. Venayagamoorthy, "Comparison of feedforward and feedback neural network architectures for short term wind speed prediction," in *Proceedings of the International Joint Conference on Neural Networks (IJCNN '09)*, pp. 3335–3340, Atlanta, Ga, USA, June 2009.
- [3] J. Wu, S. Chen, J. Zeng, and L. Gao, "Control technologies in distributed generation system based on renewable energy," *Asian Power Electronics Journal*, vol. 3, no. 1, pp. 39–52, 2009.
- [4] A. Y. Alanis, L. J. Ricalde, and E. N. Sanchez, "High Order Neural Networks for wind speed time series prediction," in *Proceedings of the International Joint Conference on Neural Networks (IJCNN '09)*, pp. 76–80, Atlanta, Ga, USA, June 2009.
- [5] T. G. Barbounis, J. B. Theocharis, M. C. Alexiadis, and P. S. Dokopoulos, "Long-term wind speed and power forecasting using local recurrent neural network models," *IEEE Transactions on Energy Conversion*, vol. 21, no. 1, pp. 273–284, 2006.
- [6] T. H. M. El-Fouly, E. F. El-Saadany, and M. M. A. Salama, "One day ahead prediction of wind speed and direction," *IEEE Transactions on Energy Conversion*, vol. 23, no. 1, pp. 191–201, 2008.
- [7] G. N. Kariniotakis, G. S. Stavrakakis, and E. F. Nogaret, "Wind power forecasting using advanced neural networks models," *IEEE Transactions on Energy Conversion*, vol. 11, no. 4, pp. 762–767, 1996.
- [8] T. Senjyu, A. Yona, N. Urasaki, and T. Funabashi, "Application of recurrent neural network to long-term-ahead generating power forecasting for wind power generator," in *Proceedings of the IEEE PES Power Systems Conference and Exposition (PSCE '06)*, pp. 1260–1265, November 2006.
- [9] Y.-K. Wu and J.-S. Hong, "A literature review of wind forecasting technology in the world," in *Proceedings of the IEEE Lausanne POWERTECH*, pp. 504–509, July 2007.
- [10] I.-H. Li, W.-Y. Wang, S.-F. Su, and Y.-S. Lee, "A merged fuzzy neural network and its applications in battery state-of-charge estimation," *IEEE Transactions on Energy Conversion*, vol. 22, no. 3, pp. 697–708, 2007.
- [11] L. J. Ricalde, G. A. Catzin, A. Y. Alanis, and E. N. Sanchez, "Higher order wavelet neural networks with Kalman learning for wind speed forecasting," in *Proceedings of the IEEE Symposium on Computational Intelligence Applications in Smart Grid (CIASG '11)*, Symposium Series on Computational Intelligence, pp. 55–60, Paris, France, April 2011.
- [12] R. J. Williams and D. Zipser, "A learning algorithm for continually running fully recurrent neural networks," *Neural Computation*, vol. 1, pp. 270–280, 1989.
- [13] C.-S. Leung and L.-W. Chan, "Dual extended Kalman filtering in recurrent neural networks," *Neural Networks*, vol. 16, no. 2, pp. 223–239, 2003.
- [14] M. Norgaard, N. K. Poulsen, and O. Ravn, "Advances in derivative-free state estimation for nonlinear systems," Technical Report IMM-REP-1988-15, Technical University of Denmark, 2000.
- [15] A. Y. Alanis, E. N. Sanchez, and A. G. Loukianov, "Discrete-time adaptive backstepping nonlinear control via high-order neural networks," *IEEE Transactions on Neural Networks*, vol. 18, no. 4, pp. 1185–1195, 2007.
- [16] L. A. Feldkamp, D. V. Prokhorov, and T. M. Feldkamp, "Simple and conditioned adaptive behavior from Kalman filter trained recurrent networks," *Neural Networks*, vol. 16, no. 5-6, pp. 683–689, 2003.
- [17] K. E. Parsopoulos and M. N. Vrahatis, *Particle Swarm Optimization*, IGI Global, 2010.
- [18] R. Kiran, S. R. Jetti, and G. K. Venayagamoorthy, "Online training of a generalized neuron with particle swarm optimization," in *Proceedings of the International Joint Conference on Neural Networks 2006 (IJCNN '06)*, pp. 5088–5095, Vancouver, Canada, July 2006.
- [19] T. Su, J. Jhang, and C. Hou, "A hybrid artificial neural networks and particle swarm optimization for function approximation," *International Journal of Innovative Computing, Information and Control*, vol. 4, no. 9, pp. 2363–2374, 2008.
- [20] C.-J. Lin and H.-M. Tsai, "FPGA implementation of a wavelet neural network with particle swarm optimization learning," *Mathematical and Computer Modelling*, vol. 47, no. 9-10, pp. 982–996, 2008.

- [21] C.-J. Lin, C.-H. Chen, and C.-T. Lin, "A hybrid of cooperative particle swarm optimization and cultural algorithm for neural fuzzy networks and its prediction applications," *IEEE Transactions on Systems, Man and Cybernetics C*, vol. 39, no. 1, pp. 55–68, 2009.
- [22] L. Cao, "Practical method for determining the minimum embedding dimension of a scalar time series," *Physica D*, vol. 110, no. 1-2, pp. 43–50, 1997.
- [23] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, pp. 1942–1948, December 1995.
- [24] A. Lazinica, *Particle Swarm Optimization*, In-Tech, 2009.
- [25] Y. Shi and R. Eberhart, "Modified particle swarm optimizer," in *Proceedings of the IEEE International Conference on Evolutionary Computation (ICEC '98)*, pp. 69–73, May 1998.
- [26] M. Clerc, "The swarm and the queen: towards a deterministic and adaptive particle swarm optimization," in *Proceedings of the Congress on Evolutionary Computation*, pp. 1951–1957, 1999.
- [27] B. Al-kazemi and C. K. Mohan, "Multi-phase generalization of the particle swarm optimization algorithm," in *Proceedings of the Congress on Evolutionary Computation*, vol. 1, pp. 489–494, 2002.
- [28] M. B. Kennel, R. Brown, and H. D. I. Abarbanel, "Determining embedding dimension for phase-space reconstruction using a geometrical construction," *Physical Review A*, vol. 45, no. 6, pp. 3403–3411, 1992.
- [29] A. Y. Alanis, C. Simetti, L. J. Ricalde, and F. Odone, "A wind speed neural model with particle swarm optimization kalman learning," in *Proceedings of the 9th International Symposium on Intelligent Automation and Control*, Puerto Vallarta, Mexico, June 2012.
- [30] National Renewable Energy Laboratory's National Wind Technology Center, <http://www.nrel.gov/wind/>.

## Research Article

# A Novel Method for Surface Defect Detection of Photovoltaic Module Based on Independent Component Analysis

Xuewu Zhang, Hao Sun, Yun Zhou, Ji Xi, and Min Li

*Key Laboratory of Sensor Networks and Environmental Sensing, Hohai University, Changzhou 213022, China*

Correspondence should be addressed to Xuewu Zhang; [lab\\_112@126.com](mailto:lab_112@126.com)

Received 22 February 2013; Accepted 7 May 2013

Academic Editor: Vishal Bhatnagar

Copyright © 2013 Xuewu Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper proposed a new method for surface defect detection of photovoltaic module based on independent component analysis (ICA) reconstruction algorithm. Firstly, a faultless image is used as the training image. The demixing matrix and corresponding ICs are obtained by applying the ICA in the training image. Then we reorder the ICs according to the range values and reform the de-mixing matrix. Then the reformed de-mixing matrix is used to reconstruct the defect image. The resulting image can remove the background structures and enhance the local anomalies. Experimental results have shown that the proposed method can effectively detect the presence of defects in periodically patterned surfaces.

## 1. Introduction

In recent years, developments in image processing, computer vision, artificial intelligence, and other related fields have significantly improved capability of visual inspection techniques. The objective of this paper is to explore and research effective algorithms in automatic inspection system of photovoltaic module defects.

Many products surfaces found in manufacturing have regularly patterned surfaces and can be classified as structural textures in images. The automated visual inspection in textured surfaces is then to detect small and ill-defined defects that locally break the homogeneity of a texture pattern. Numerous methods have been proposed to extract textural features either directly from spatial domain or from the spectral domain. The gray-level cooccurrence matrix (GLCM) is one of the statistic methods in the spatial domain; Zou et al. [1] present a method of Fuzzy Label Cooccurrence Matrix (FLCM) set to detect defects of colour fabric, which enhanced the computation performance in real-time application. Typically, the features in spectral domain are generally less sensitive to noise than features in spatial domain due to the relatively uniform representation in the spectral domain. Techniques in the spectral domain extract textural features by

conducting frequency transforms such as Fourier transforms, Gabor transforms, or wavelet transforms. Fourier transform is a global method, which just depicts the spatial-frequency distribution without regarding to the spatial domain information [2–4]. Gabor transform belongs to windowing Fourier transform. Gabor function has well frequency description capability, which is similar to biological effects of human eye and can be used to extract the corresponding feature from different size and different direction in frequency domain. Gabor filters extract features by filtering the textured image with a set of Gabor filter banks characterized by the frequency, the sinusoid orientation, and the Gaussian function scale. However, for Gabor transform is nonorthogonal, redundancy exists in different feature components, which results in computationally intensive in analyzing texture image [5]. Recently, wavelet transform provides a convenient way to obtain a multiresolution representation, from which texture features are easily extracted. It has been a popular alternative for the extraction of textural features and has been successfully applied for texture segmentation and defect detection [6].

In recent years, independent component analysis has attracted a lot of attention in image processing applications. The basic idea in the ICA model is to construct basis images

for a given image result in weighting components which are mutually statistically independent [7]. ICA filters are finite support filters whose size is determined by the size of the image patches selected. They are data independent in that they have been constructed from the available training data. A few studies have also been reported in the use of ICA for texture analysis. Sezer et al. [8] proposed an ICA-based defect detection method for textile fabric images. The method makes use of independent component analysis for feature extraction from the nonoverlapping subwindows of texture images. Euclidean distance between the feature obtained from average value of the feature of a defect free sample and the feature obtained from one subwindow of a test image is used to classify a subwindow as defective or nondefective. Tsai and Lai [9] proposed a fast self-comparison scheme for defect detection in structural surfaces containing periodic complicated patterns. The scheme is simply carried out by dividing a sensed line image into two segments of equal length. An independent component analysis model is proposed to obtain the demixing matrix that can recover the translation between the two divided segments. The normalized cross-correlation is adopted to measure the similarity between two compared segments. Experimental results have shown that the proposed self-comparison scheme can effectively and efficiently detect the presence of defects in periodically patterned surfaces. Jenssen and Eltoft [10, 11] presented independent component analysis of textured images as a computational technique for creating a new data dependent filter bank for use in texture segmentation.

At present, the researches mostly focus on sensory system, including scanning acoustic microscopy (SAM) [12], optical transmission [13], ultrasound lock-in thermography [14], luminescence [15, 16], and resonance ultrasound vibration [17]. The SAM [12] method of crack assessment is not feasible for mass production of photovoltaic cells because the time required to scan a 100 mm by 100 mm wafer is between 10 and 15 min. Additionally, the wafer has to be submerged in a water bath or covered with a water droplet. However, this approach does allow cracks as small as 5–10  $\mu\text{m}$  to be detected. The optical crack detection system utilizes the transmission of a high intensity flashlight through the wafer and captures the image with a CCD camera coupled with optical filters [13]. To decrease the long measurement periods, infrared- (IR-) camera lifetime mapping/carrier density imaging (ILM/CDI) has been introduced recently [14]. Using an IR camera (i.e., an array of IR detectors) sensitive in the wavelength range between  $\sim 3$  and  $8 \mu\text{m}$  instead of a single detector reduces the measurement period for a high-resolution mapping from hours to minutes and sometimes even seconds. More recently, camera-based photoluminescence imaging has been introduced, which also allows a very fast imaging of silicon wafers and solar cells at high resolution using a conventional relatively cheap silicon CCD camera [15, 16]. Though luminescence methods are fast and non-destructive, other types of defects such as surface scratches and dislocations may interfere and misinterpret the crack identification.

This paper concentrates on research issues related to defect classification in solar cell module and discussed the

general methodology as well as specific examples of the algorithms. The aim of this paper is to illustrate the potential of ICA in solar cell module defects detection. In this study, the proposed ICA-based method uses an image reconstruction strategy to eliminate the repetitive structural pattern of a solar cell surface.

The paper is organized as follows: in Section 2, firstly the basic ICA model is overviewed, and then the ICA model used for this paper is described in detail. Defect reconstruction procedure using ICA is presented in Section 3. Section 4 showed the experimental results and discussed the performance of these algorithms; Section 5 summarizes the results.

## 2. Methodology

*2.1. Basic ICA Model.* ICA has emerged as one powerful solution to the problem of blind source separation and has attracted broad attention.

We represent an image as a column vector  $X = [x_1, \dots, x_M]^T$ , which is modeled as linear combination of  $n$ -dimensional variables  $s$ , as shown in (1):

$$X = \sum_{i=1}^N s_i a_i = AS. \quad (1)$$

$S = [s_1, \dots, s_M]^T$  (often described as independent components, ICs) is latent variables that cannot be directly observed, where the basis function  $a_i$ ,  $i = 1, \dots, N$ , are the columns of the  $(M \times N)$  matrix  $A$ . We usually assume  $S$  is statistically independent and the mixing matrix  $A$  is unknown.

In order to learn the matrix  $A$ , we attempt to find a linear transformation  $W$  of the training data  $X$ , which yields a vector

$$WX = Y, \quad (2)$$

which results in components of  $Y$  being as statistically independent as possible. The vector  $Y$  is an estimate of  $S$ . Matrix  $A$  is found as the pseudoinverse of  $W$ .

There exist several iterative algorithms performing ICA. In this research, we used FastICA algorithm proposed by Hyvarinen according to the approximation negentropy.

*2.2. Preprocessing.* Before applying an ICA algorithm on the images, it is usually very useful and necessary to do some preprocessing. The most basic preprocessing is to make  $X$  mean zero, by subtracting  $X$  by its mean. This course is called centering. Another useful preprocessing in ICA is to make each image matrix unit covariance; this course is called whitening. In this processing,  $X$  is multiplied by  $ED^{-1/2}E^T$ , where  $E$  is the orthogonal matrix of eigenvectors of  $E(XX^T)$  and  $D$  is the diagonal matrix of its eigenvalue.

## 3. ICA Reconstruction Used in Defect Inspection

The detailed procedure of the proposed scheme is summarized as follows.

### Training

- (1) Select a faultless image of size  $256 \times 256$  as the training image, and center and whiten the image.
- (2) Compute the demixing matrix and independent components, using the fast-ICA algorithm.
- (3) Sort the ICs in decreasing order and reorder the ICs.
- (4) Select the number of IC and the corresponding  $W_i$ .
- (5) Reform the demixing matrix  $W$ , and reconstruct the image  $X$ .

### Inspection

- (6) Center and whiten the inspection image  $X_I$  of size  $N \times M$ .
- (7) Reconstruct the inspection image  $X_I$ .
- (8) Binarize the reconstructed source image using the control limits:

$$y_{ij} = \begin{cases} 255 \text{ (faultless point),} & \\ \mu_i - t \cdot \sigma_i < y_{ij} < \mu_i + t \cdot \sigma_i, & \text{for } i = 1, 2, \dots, N; \\ & j = 1, 2, \dots, M \\ 0 \text{ (defective point),} & \text{otherwise,} \end{cases} \quad (3)$$

where  $y_{ij}$  is the gray level of image at coordinate  $(i, j)$ ,  $\mu_i$  and  $\sigma_i$  are the mean and standard deviations of the gray values in the  $i$ th row of image, and  $t$  is a control constant.

For each row image,  $\mu_i - t \cdot \sigma_i$  and  $\mu_i + t \cdot \sigma_i$  are the upper and lower control limits of intensity, respectively.

## 4. Experimental Results and Analysis

The experiments were conducted on a Pentium 4 1.8 GHz personal computer processor. The FastICA package is applied to generate the demixing matrix of each training image.

**4.1. Gray-Level Properties of Solar Cell Panel.** A photovoltaic module is the basic element of each photovoltaic system. It consists of many jointly connected solar cells. The most commercial crystalline modules consist of 36 or 72 cells. In this paper, we have 72 cells. Every cell has size of 125 mm  $\times$  125 mm; so the width of defect should be above 1.5 mm. Solar cells are connected and placed between a Tedlar plate on the bottom and a tempered glass on the top. Solar cells are interconnected with thin contacts on the upper side of the semiconductor material, which can be seen as a metal net on the solar cells. The net must be as thin as possible allowing a disturbance free incidence photon stream. Usually a module is framed with an aluminum frame, occasionally with a stainless steel or with a plastic frame. Figure 1 shows a faultless solar module image in a coarse resolution of 40  $\times$  967 pixels. We can see that Figure 1 has structural texture surfaces that involve only simple periodic patterns.

To further analyze the characteristics of solar cell images, we show solar cell image in fine resolution. Figures 2(a), 2(b), and 2(c) show three 2D images that contain a particle

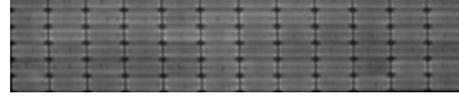


FIGURE 1: A faultless solar cell image of size 40  $\times$  967 pixels.

defect in different areas of solar cell in a fine resolution. The line images in Figures 2(g)–2(i) are horizontal gray-level profiles of defective regions in Figures 2(a)–2(c) and Figures 2(d)–2(f) horizontal gray-level profiles of faultless regions corresponding to Figures 2(a)–2(c), respectively. A gray-level profile in the pixel area has the most simple and regular pattern, as seen in Figure 2(d). The pattern associated with the particle defect in one period is different to some extent from the normal one.

We can see that photovoltaic module image contains a highly repetitive pattern in a very short period.

Consider a faultless training 2D image as a data matrix for the ICA model. Each row vector of the matrix is treated as an observed mixture signal. Then, each row vector of the estimated source matrix obtained from the ICA model represents an independent source signal that describes a unique edge of the patterned structure in the surface. All these independent source signals compose the complete structure of a patterned solar cell panel used in training.

**4.2. IC Properties of Solar Module.** As shown in Figure 1, it contains 40 row images, each of size 1  $\times$  967 pixels. By applying ICA to the image, 40 independent components, each of size 1  $\times$  967, can be obtained. Figure 3 depicts the profiles of the resulting 40 ICs.

From Figure 3, we can see that some ICs include visible spiky points while others just have flat profiles. The ICs with spiky points can be used to represent the global structural information of a patterned solar module image. Conversely, the ICs with flat profiles can be used to represent the local uniform information of a patterned solar module image.

When using ICA for defect detection, we can first identify the proper ICs that contain significant spiky points to represent the main structure of the image. The corresponding  $W_i$  of such identified ICs are then removed and replaced with a specific  $W_k$  that shows the flattest IC profile to reform the demixing matrix  $W$ . By using the reformed  $W'$  to reconstruct the inspection image, the global background texture will be removed and the local anomalies will be presented in the reconstructed image.

The ICs can be taken as basis images to represent the feature of the input image (the ICs of an image are well suited to describe the local feature variation of the image). When using a faultless patterned solar module image as an input matrix to the ICA model, the profiles of ICs will either have a flat structure or involve obvious spiky points.

**4.3. ICA Reconstruction of Solar Module Image.** Compute the difference between max value and min value for each IC, and then sort the ICs in decreasing order and reorder the ICs. The new ICs are shown in Table 1.

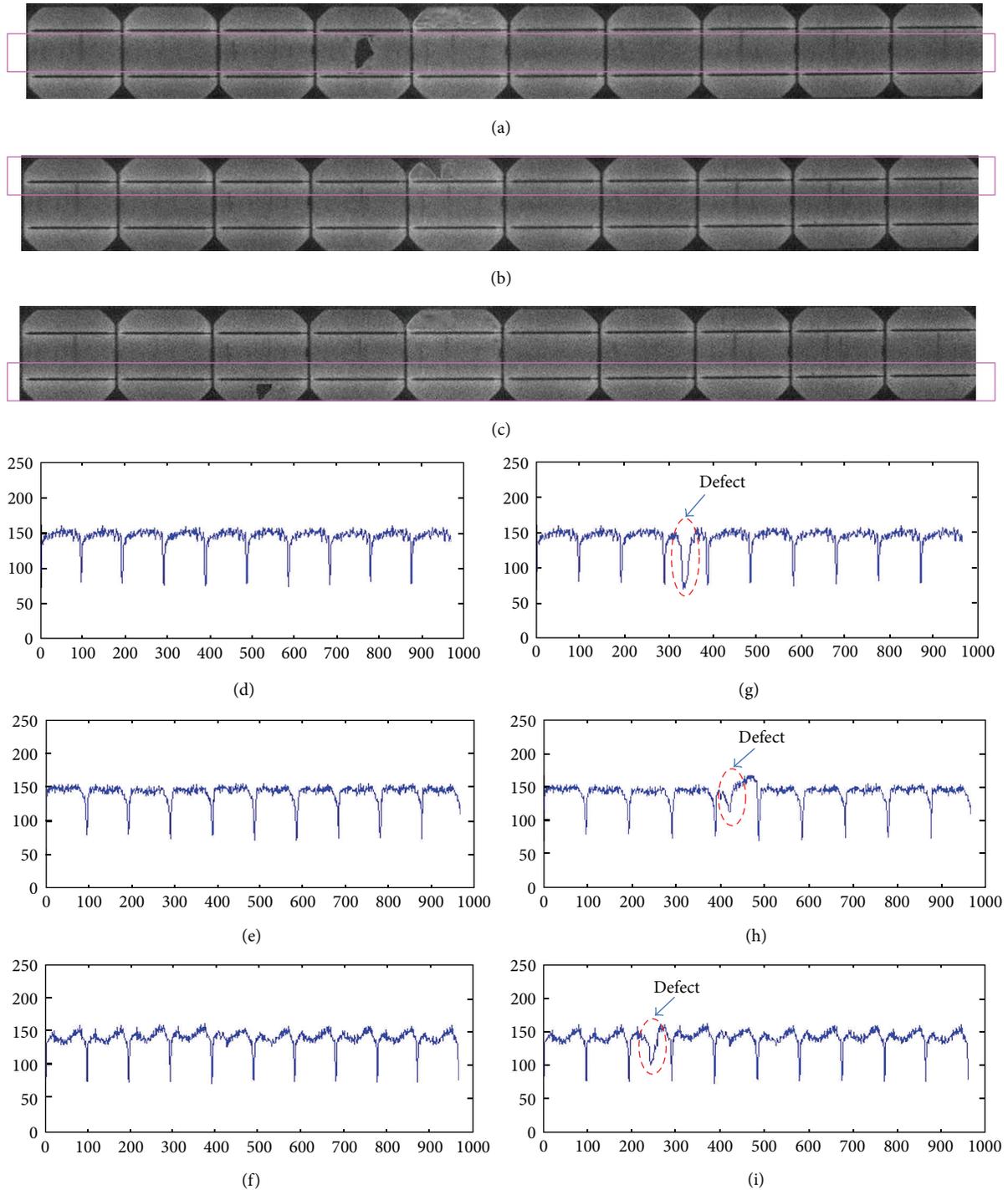


FIGURE 2: Gray profiles of solar cell panel.

Since ICA can only recover the source signals up to permutation (and scaling), the order of the 40 ICs in Figure 4 does not correspond to the row sequence in the original image  $X$ . It can be seen from Figure 4 that the profiles of some ICs contain visible spiky points; for example, IC2 is one of the examples. The ICs that contain obvious spiky points compose the main structural pattern of the solar cell panel image. It can also be found from Figure 3 that some ICs

have near uniform profiles; for example, IC12 is one of the examples. Those ICs can be used to describe the uniform (nonstructural) background of the solar cell image.

In this paper, symmetric fixed-point ICA algorithm with  $\tanh(x)$  nonlinearity is used [15].

By means of ICA, hidden factors underlying the fabric images data set are obtained. Sixteen ICs are used for the analysis. This value is obtained by trial and error.

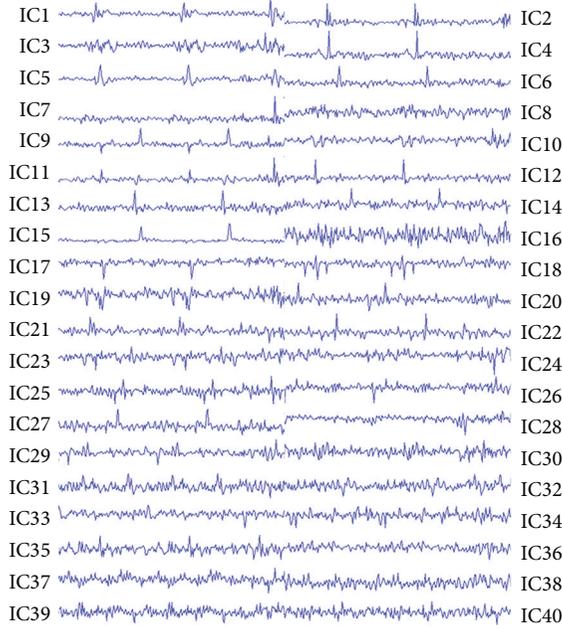


FIGURE 3: The profiles of the resulting 40 ICs.

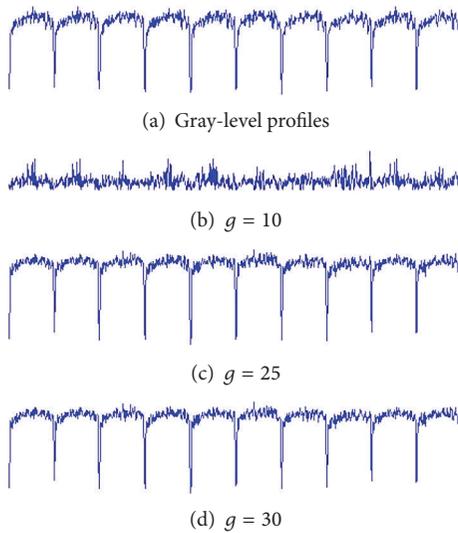


FIGURE 4: Reconstruction of solar cell image using different numbers of ICs with obvious spiky points: (a) original gray-level profile of the row image in Figure 6(a); (b)~(d) reconstructed row images using 10, 25, and 30 ICs, respectively.

It can be seen that IC5 has the largest range value and IC39 has the smallest range value. As mentioned in Table 1, the selection number of the replaced ICs is 25, that is,  $IC_1', IC_2', \dots, IC_{25}'$ , and the replacing IC is  $IC_{40}'$ . Based on the replaced ICs and the replacing IC, the demixing matrix  $W$  is reformed to a new demixing matrix  $W^*$ .

Figures 4 and 5 show the gray-level profiles of reconstruct defective image and faultless image using the new demixing matrix, respectively. It can be observed that the profile of the reconstructed row image gets closer to that of the original row

TABLE 1: ICs value.

IC	$R_i$
$IC_5(IC_1')$	16.2770
$IC_1(IC_2')$	16.0817
$IC_{10}(IC_3')$	15.5875
$IC_3(IC_4')$	14.3378
$IC_2(IC_5')$	14.0981
$IC_8(IC_6')$	12.9055
$IC_{11}(IC_7')$	12.8781
$IC_{16}(IC_8')$	12.7694
$IC_7(IC_9')$	12.4899
$IC_4(IC_{10}')$	11.9651
$IC_{23}(IC_{11}')$	11.2746
$IC_{20}(IC_{12}')$	10.7203
$IC_9(IC_{13}')$	10.6168
$IC_{12}(IC_{14}')$	10.6159
$IC_{21}(IC_{15}')$	10.4436
$IC_{17}(IC_{16}')$	10.1953
$IC_6(IC_{17}')$	10.0291
$IC_{25}(IC_{18}')$	9.9173
$IC_{19}(IC_{19}')$	9.5628
$IC_{24}(IC_{20}')$	9.4935
$IC_{22}(IC_{21}')$	9.4757
$IC_{13}(IC_{22}')$	9.3286
$IC_{28}(IC_{23}')$	9.1619
$IC_{14}(IC_{24}')$	9.0582
$IC_{30}(IC_{25}')$	8.8652
$IC_{29}(IC_{26}')$	8.8194
$IC_{32}(IC_{27}')$	8.7825
$IC_{26}(IC_{28}')$	8.7541
$IC_{33}(IC_{29}')$	8.5220
$IC_{31}(IC_{30}')$	8.4803
$IC_{35}(IC_{31}')$	8.4722
$IC_{36}(IC_{32}')$	8.4665
$IC_{18}(IC_{33}')$	8.4331
$IC_{27}(IC_{34}')$	8.2911
$IC_{40}(IC_{35}')$	8.1047
$IC_{38}(IC_{36}')$	8.1013
$IC_{34}(IC_{37}')$	8.0307
$IC_{15}(IC_{38}')$	7.9053
$IC_{39}(IC_{39}')$	7.8023
$IC_{37}(IC_{40}')$	7.2717

image as the number of spiky ICs is increased. Note that 25 out of a total of 40 ICs are enough to reconstruct the main structure of the original row image, as shown in Figures 4 and 5.

Figure 6 shows the typical samples of solar cell module. Figure 7 shows the reconstructed images by using the newly

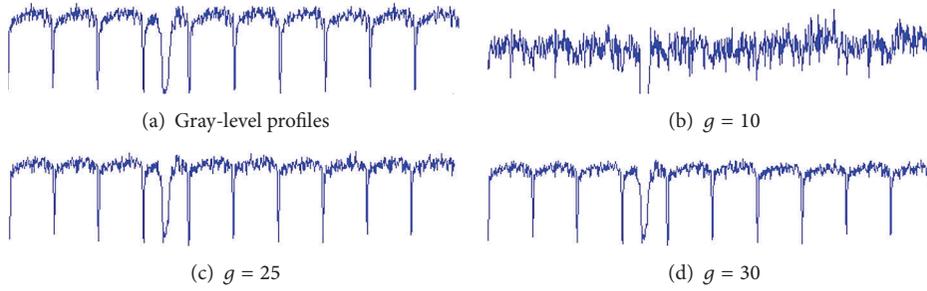


FIGURE 5: Reconstruction of solar cell image using different numbers of ICs with obvious spiky points: (a) original gray-level profile of the row image in Figure 5(a); (b)~(d) reconstructed row images using 10, 25, and 30 ICs, respectively.

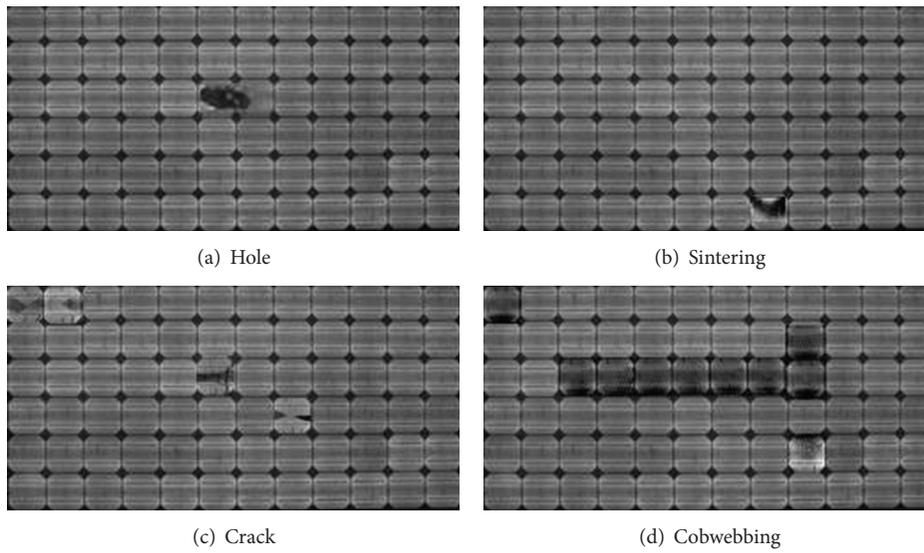


FIGURE 6: Typical samples.

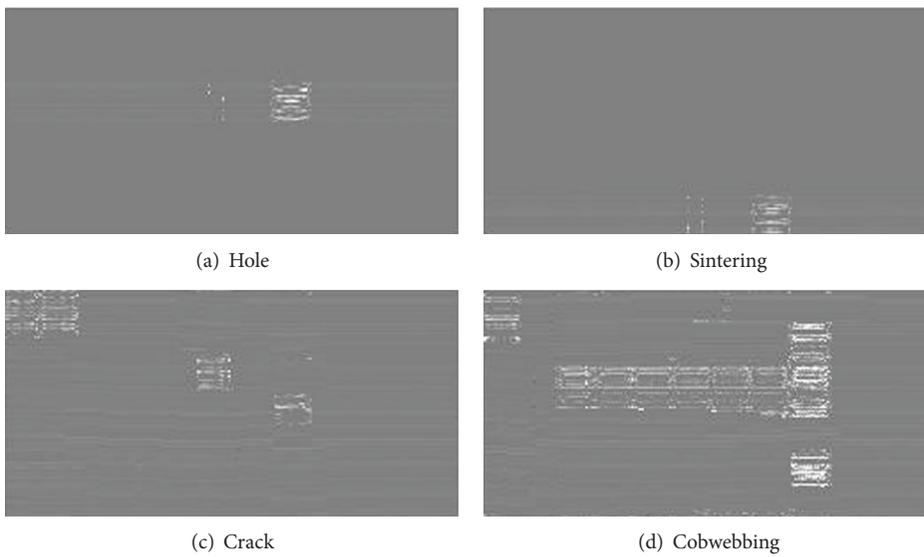


FIGURE 7: Reconstruction results using ICA.

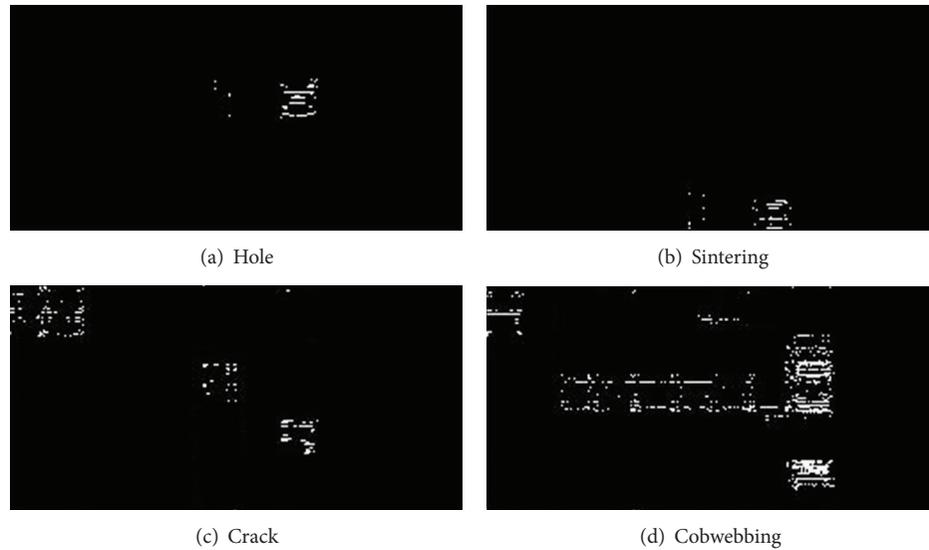


FIGURE 8: Binary results.

reformed demixing matrix to reconstruct the test images in Figure 6, and Figure 8 shows the binary results. We can see that, for defective images, the background texture is effectively eliminated and defects are well segmented. For the faultless image, the resulting binary image is uniformly white and no defect is claimed.

If the gray value  $k$  falls with the control limits, it is classified as a faultless point of the inspection surface. Otherwise, it is classified as a defective point. The constant  $t$  is taken by 2.5 here.

## 5. Conclusions

In this study, an ICA-based system for inspection of photovoltaic module defects was developed. A photovoltaic module consists of many jointly connected solar cells, so it can be processed as repetitive patterns. A faultless image is used as the training image. The demixing matrix and corresponding ICs are obtained by applying the ICA in the training image. Then we re-order the ICs according to the range values and reform the demixing matrix. Then the reformed demixing matrix is used to reconstruct the defect image. The resulting image can remove the background structures and enhance the local anomalies. Experimental results have shown that the proposed method can effectively detect the presence of defects in periodically patterned surfaces. This method correctly locates the position of defects but does not reflect the true shape of the defects, and the length of defect size is above 1.5 mm.

## Acknowledgments

This paper is supported by the National Natural Sciences Foundation of China (Grant no. 61273170); Ph.D. Programs Foundation of the Ministry of Education of China

(20120094120023); Changzhou Science and technology Support Program (Industrial) (no. CE20120101). Xuewu Zhang is an Associate Professor of communication engineering at the College of Computer and Information, Hohai University. He received the B.S. degree, the M.S. degree, and the Ph.D. degree from the Computer and Information College, Hohai University in 1998, 2006, and 2011, respectively. He has published over 40 research papers and obtained over 10 patents. His research interests include pattern recognition, multimedia signal processing, machine learning, and bionics.

## References

- [1] C. Zou, B. W. Wang, and Z. G. Sun, "Detection of fabric defects with Fuzzy Label Co-occurrence Matrix set," *Journal of Donghua University*, vol. 26, no. 5, pp. 549–553, 2009.
- [2] H. Y. T. Ngan, G. K. H. Pang, S. P. Yung, and M. K. Ng, "Wavelet based methods on patterned fabric defect detection," *Pattern Recognition*, vol. 38, no. 4, pp. 559–576, 2005.
- [3] X. W. Zhang, L. Z. Xu, Y. Q. Ding, X. N. Fan, L. P. Gu, and H. Sun, "Inspection of surface defects in copper strip based on machine vision," *Life System Modeling and Intelligent Computing*, vol. 6328, no. 1, pp. 304–312, 2010.
- [4] S. C. Kim and T. J. Kang, "Texture classification and segmentation using wavelet packet frame and Gaussian mixture model," *Pattern Recognition*, vol. 40, no. 4, pp. 1207–1221, 2007.
- [5] O. Pichler, A. Teuner, and B. J. Hosticka, "A comparison of texture feature extraction using adaptive Gabor filtering, pyramidal and tree structured wavelet transforms," *Pattern Recognition*, vol. 29, no. 5, pp. 733–742, 1996.
- [6] T. S. Li, "Applying wavelets transform and support vector machine for copper clad laminate defects classification," *Computers and Industrial Engineering*, vol. 56, no. 3, pp. 1154–1168, 2009.
- [7] A. Hyvärinen and E. Oja, "Independent component analysis: algorithms and applications," *Neural Networks*, vol. 13, no. 4–5, pp. 411–430, 2000.

- [8] O. G. Sezer, A. Ertuzun, and A. Ercil, "Independent component analysis for texture defect detection," *Pattern Recognition and Image Analysis*, vol. 14, no. 2, pp. 303–307, 2004.
- [9] D. M. Tsai and S. C. Lai, "Defect detection in periodically patterned surfaces using independent component analysis," *Pattern Recognition*, vol. 41, no. 9, pp. 2812–2832, 2008.
- [10] R. Jenssen and T. Eltoft, "ICA filter bank for segmentation of textured images," in *Proceedings of the 4th International Symposium on Independent Component Analysis and Blind Signal Separation (ICA '03)*, Nara, Japan, 2003.
- [11] R. Jenssen and T. Eltoft, "Independent component analysis for texture segmentation," *Pattern Recognition*, vol. 36, no. 10, pp. 2301–2315, 2003.
- [12] A. Belyaev, O. Polupan, S. Ostapenko, D. Hess, and J. P. Kalejs, "Resonance ultrasonic vibration diagnostics of elastic stress in full-size silicon wafers," *Semiconductor Science and Technology*, vol. 21, no. 3, pp. 254–260, 2006.
- [13] E. Rueland, A. Herguth, A. Trummer, S. Wansleben, and P. Fath, "l-Crack detection and other optical characterization techniques for in-line inspection of wafers and cells," in *Proceedings of 20th EU PVSEC*, pp. 3242–3245, Barcelona, Spain, 2005.
- [14] J. P. Rakotoniaina, O. Breitenstein, M. H. Al Rifai, D. Franke, and A. Schnieder, "Detection of cracks in silicon wafers and solar cells by lock-in ultrasound thermography," in *Proceedings of PV Solar conference*, pp. 640–643, Paris, France, 2004.
- [15] T. Trupke, R. A. Bardos, M. C. Schubert, and W. Warta, "Photoluminescence imaging of silicon wafers," *Applied Physics Letters*, vol. 89, no. 4, Article ID 044107, 2006.
- [16] T. Fuyuki, H. Kondo, T. Yamazaki, Y. Takahashi, and Y. Uraoka, "Photographic surveying of minority carrier diffusion length in polycrystalline silicon solar cells by electroluminescence," *Applied Physics Letters*, vol. 86, no. 26, Article ID 262108, pp. 1–3, 2005.
- [17] A. Belyaev, O. Polupan, W. Dallas, S. Ostapenko, D. Hess, and J. Wohlgemuth, "Crack detection and analyses using resonance ultrasonic vibrations in full-size crystalline silicon wafers," *Applied Physics Letters*, vol. 88, no. 11, Article ID 111907, 2006.

## Research Article

# Weight-Selected Attribute Bagging for Credit Scoring

**Jianwu Li, Haizhou Wei, and Wangli Hao**

*Beijing Laboratory of Intelligent Information Technology, School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100081, China*

Correspondence should be addressed to Jianwu Li; [ljw@bit.edu.cn](mailto:ljw@bit.edu.cn)

Received 21 March 2013; Accepted 29 April 2013

Academic Editor: Yudong Zhang

Copyright © 2013 Jianwu Li et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Assessment of credit risk is of great importance in financial risk management. In this paper, we propose an improved attribute bagging method, weight-selected attribute bagging (WSAB), to evaluate credit risk. Weights of attributes are first computed using attribute evaluation method such as linear support vector machine (LSVM) and principal component analysis (PCA). Subsets of attributes are then constructed according to weights of attributes. For each of attribute subsets, the larger the weights of the attributes the larger the probabilities by which they are selected into the attribute subset. Next, training samples and test samples are projected onto each attribute subset, respectively. A scoring model is then constructed based on each set of newly produced training samples. Finally, all scoring models are used to vote for test instances. An individual model that only uses selected attributes will be more accurate because of elimination of some of redundant and uninformative attributes. Besides, the way of selecting attributes by probability can also guarantee the diversity of scoring models. Experimental results based on two credit benchmark databases show that the proposed method, WSAB, is outstanding in both prediction accuracy and stability, as compared to analogous methods.

## 1. Introduction

The assessment of credit risk has become increasingly crucial for financial institutions because high risks associated with inappropriate credit decisions may result in great losses [1]. In the latest financial crisis, many financial institutions suffered heavy losses from numerous customers' defaults on loans. Therefore, effective methods for evaluating credit risk are needed to help financial institutions to avoid losses [2]. The objective of credit scoring is to assign credit applicants to either a "good credit" group with high possibility to repay their financial obligation or a "bad credit" group with high possibility to default on their financial obligation. When considering the application for a large loan, the lender tends to evaluate the risk by a loan officer or even a committee to investigate the applicant in detail. Nevertheless, with the rapid growth and increasing competition in credit industry, it is necessary to perform fast automatic decisions on credit risk evaluations for financial institutions, especially when facing millions of applications for credit cards or consumer loans simultaneously. This demand leads to the birth of quantitative credit scoring method.

Quantitative credit scoring has gained more and more attention in recent years because an improvement in accuracy, even a fraction of a percent, can translate into significant future savings for the credit institutions [3]. Quantitative credit scoring models are developed based on the observations of historical data including income, age, and profession for good and bad examples, respectively. An excellent model should allow accurate classification of new applicants as good or bad.

Numerous models have been developed to evaluate consumer loans and improve credit scoring accuracy [4]. Initially, some statistical techniques are widely used to build credit scoring models such as linear discriminate analysis (LDA) [5] and logistic regression (LR) [6]. Although the two methods are relatively simple and explainable, the ability to discriminate good credit applicants from bad ones is still disputed. LDA is criticized because it needs a strong hypothesis such as the categorical property of the data and the variance homogeneity. In reality, the covariance matrices of good credit data are considerably distinct from those of bad credit data. Besides, both LDA and LR are linear classifiers, such that they are not suitable for complex

nonlinear classification problems featured in credit scoring. In recent years, some new methods from the field of artificial intelligence have also been applied to the credit scoring, such as decision trees [7, 8],  $k$ -nearest neighbor (KNN) [9], artificial neural networks [10, 11], genetic algorithms [12–14], genetic programming [15, 16], artificial immune algorithms [17], and support vector machines (SVM) [18, 19]. Among these artificial intelligence methods, decision trees, artificial neural networks, and support vector machines are generally regarded as the most efficient individual models [20]. Furthermore, in order to improve the prediction accuracy and overcome the shortcoming of individual scoring model, two-stage scoring models [16, 21], hybrid scoring models [22, 23] and ensemble scoring models [20, 24] are also introduced. Experimental results show that these models perform better than individual classifiers.

Ensemble learning that combines outputs from multiple individual classifiers is one of the most important techniques for improving classification accuracy in machine learning [25, 26]. For example, an ensemble of multiple least square SVMs can obtain higher accuracy than the individual least square SVM in credit scoring and bankruptcy prediction [24]. Among ensemble learning models, bagging (short for “bootstrap aggregating”) and boosting are two kinds of popular and widely used methods. Standard bagging (SB) [27, 28] is based on data partitioning which produces  $k$  working sets, each one with the same size as the original training set through randomly sampling from the original training set with replacement. Then, each working set is used to train a child classifier independently. During test phase, a test instance is evaluated by all child classifiers simultaneously and a collective decision is obtained based on some aggregation strategy. Boosting [25, 26, 29] is also based on data partitioning. Boosting produces a series of child classifiers, and the training dataset of each child classifier is generated according to the classification errors of previously created child classifiers. Test examples are classified by combining the predictions of all child classifiers according to a special aggregation strategy. AdaBoost is the most frequently used boosting method [25, 26].

Theoretical and experimental results suggest that combining classifiers can give effective improvement in accuracy if classifiers within an ensemble are not correlated with each other [30, 31]. One of the most effective methods of achieving such independence is to train the members in the ensemble by using different attribute subsets [32]. In other words, attribute partitioning methods can obtain better performance than data partitioning methods in ensemble learning [30]. Ensemble learning methods based on attribute partitioning are called as attribute bagging (AB) and have been investigated in many publications [33, 34]. Also, some AB models have been used to construct credit scoring systems and show promising results [35, 36].

For attribute bagging models, the selection of optimal attribute subsets plays an important role. Usually, attributes are selected randomly to construct attribute subsets. This method is called randomly selected attribute bagging (RSAB) [30]. However, RSAB has a deficiency that elements of some attribute subsets may contribute little to classification. The

individual classifiers trained by such subsets perform badly in terms of accuracy. These individual classifiers lead to bad bagging results.

To overcome the shortcomings of RSAB, we propose a new attribute ensemble learning method, namely, weight-selected attribute bagging (WSAB). WSAB is based on the fact that some attributes are more important for the classification problem than others [37, 38]. Therefore, the more important attributes should appear more frequently in attribute subsets of AB model so as to guarantee that all individual classifiers perform well. In order to achieve this, the probabilities of the important attributes to be selected into attribute subsets should be larger than those of the unimportant attributes. Besides, the individual classifier that uses only a subset of original attributes will become more accurate after eliminating some redundant and uninformative features. On the other hand, given a certain size of attribute subsets (smaller than the total number of original attributes), selecting attributes by probability can result in some differences between different attribute subsets. Therefore, the diversity between different classifiers in an ensemble can be ensured. In other words, the WSAB can still keep the independence to some extent among different classifiers.

The implementation of the WSAB model contains two phases. In the first phase, weights of attributes need to be calculated using some attribute evaluation method. The weight  $w_i$  expresses the importance extent of the  $i$ th attribute for a given classification problem. In the second phase, an appropriate attribute subset size is firstly decided. Then, weights of attributes are used to construct attribute subsets such that the attributes with the larger weights will be selected into attribute subsets with the larger probability. In this way, attribute subsets contain the attributes that are important to classification, with the large probability, so that the accuracy of individual classifiers can be guaranteed. Then, like normal attribute bagging (AB), projections of training examples onto every attribute subset are created, respectively. Individual classifiers are trained based on every projection, and test instances are classified by combining the predictions of all individual classifiers according to a specific aggregation strategy, usually voting by majority. This paper attempts to introduce the WSAB model to solve credit scoring problem. Experimental results, based on two credit datasets from the UCI (University of California, Irvine, CA, USA) Machine Learning Repository [39], show that the WSAB is outstanding in both the prediction accuracy and stability, compared with RSAB, SB, AdaBoost, and single classifier.

The rest of this paper is organized as follows. The related research work is reviewed in Section 2. The WSAB model is described in Section 3. Subsequently, experimental results and empirical analysis are given in Section 4. The last section concludes this paper and addresses the future research task.

## 2. Related Work

*2.1. Data Partitioning Ensemble Methods.* Bagging [27] is a kind of important ensemble learning methods for improving prediction accuracy in machine learning. Standard bagging

(SB) is based on data partitioning. Given a training set  $S$  with size  $n$ , training examples are randomly sampled with replacement to generate  $m$  new sample subsets  $S_1, S_2, \dots, S_m$ , each with the same size as the original training set. Due to sampling with replacement, some training instances may be repeated several times in a new training set, and also some may not appear at all. Then, standard bagging trains  $m$  models,  $M_1, M_2, \dots, M_m$ , based on  $m$  training subsets, respectively. For a test instance, the final result is obtained via combining the outputs of  $m$  models by a specific strategy (voting for classification or averaging for regression). According to theoretical and empirical results, standard bagging can give a significant improvement in classification accuracy as well as stability [24]. In standard bagging, each classifier may be less accurate than the classifier using all training examples. However, after these classifiers are combined, the ensemble result is more accurate than the single classifier using all training samples. The diversity among individual classifiers compensates for the accuracy loss of individual classifiers in ensemble and hence improves prediction performance.

Boosting [29] is another effective method to improve the accuracy of any given learning algorithm. Boosting produces a series of classifiers, and the training dataset of each classifier is generated based on the accuracy of previously created classifiers. The samples misclassified by previously created classifiers will have larger probability to be selected into the new training dataset. By doing this, the new classifier can pay more attention to the samples that are difficult for previously created classifiers. Boosting can be implemented in several different ways. Arcing [28] and AdaBoost [40] are two important representatives. In Arcing, the classifiers' votes are weighted equally, while AdaBoost weights the predictions based on classifiers' training accuracies. It is noted that the effectiveness of boosting depends more on the data set than on base classifiers. Though boosting can significantly improve performance of weak classifiers, the ensemble is easy to focus on several special training examples that are difficult to be classified. Therefore, boosting is not stable.

Standard bagging and AdaBoost do not need too many rounds in training. Experimental results [34] show that the improvement of the performance of the learning model occurs often in the first several rounds.

*2.2. Attribute Partitioning Ensemble Methods.* Compared to data partitioning ensemble methods, attribute partitioning ensemble methods can make individual classifiers within an ensemble more "independent" [32] and then can obtain better effectiveness [30].

The bagging method based on attribute partitioning can be called attribute bagging (AB). The AB method generates attribute subsets through selecting attributes from the whole attribute set without replacement. Then, projections of training examples onto attribute subsets are created. Each child classifier is trained based on each projection, respectively, and all child classifiers are aggregated by some combination strategy. During the test phase, a test instance is fed to all child classifiers simultaneously and a collective decision is obtained based on the aggregation strategy. In conventional attribute

bagging methods, attribute subsets are generated through randomly selecting attributes from the whole attribute set. This method is called randomly selected attribute bagging (RSAB). For RSAB, all attributes have the same probability to be selected into one attribute subset. However, some attributes are very important but the others are not important for classification problems. As mentioned before, RSAB has a deficiency that some attribute subsets may only contain the attributes that contribute less to classification. Such classifiers are prone to resulting in bad bagging results.

To overcome the deficiency of RSAB, some optimization methods are used to select optimal attribute subsets. Guerra-Salcedo and Whitley use a genetic algorithm (GA) to explore the space of all possible feature subsets [41]. Their experiments compare two data partitioning ensemble methods including bagging and AdaBoost with three different AB models: complete, random and genetic, search. Experimental results show that attribute subsets selected by GA perform best, followed by RSAB. Optiz also uses GA to search for attribute subsets for ensembles [33], and experimental results also demonstrate the fact that genetic ensemble feature selection (GEFS) performs better than standard bagging (SB) and AdaBoost. However, using GA to select attribute subsets is very computationally intensive, and GA needs to evaluate each classifier after combining two objectives—accuracy and diversity in a subjective manner. Another method of selecting optimal attribute subsets for ensemble is proposed by Bryll et al. [30]. They only used the best random attribute subsets for voting. Their experiments show that ranking attribute subsets by classification accuracy and then only using the best subsets further improve the classification performance of ensemble. However, to determine which individual classifier is more accurate is very difficult because test dataset is not known beforehand. Besides, only using the best classifiers to perform ensemble may reduce the diversity among classifiers. Therefore, it is difficult to reach a balance between accuracy and diversity of individual classifiers.

### 3. Weight-Selected Attribute Bagging (WSAB)

*3.1. Evaluating Weights of Attributes.* In the first phase of WSAB modeling, weights of attributes need to be computed using attribute (or feature) evaluation method. In practice, many methods can be used to obtain weights of attributes such as linear SVM (LSVM), principal component analysis (PCA), correlation analysis, F-score model, LDA, and multivariate adaptive regression splines (MARS). Different approaches to decide weights of attributes are of different characteristics. In this paper, we attempt to employ LSVM and PCA, respectively, to evaluate weights of attributes.

*3.1.1. Evaluating Weights of Attributes via LSVM.* The SVM, proposed by Vapnik [42], is based on the statistical learning theory and has showed state-of-the-art performance for many classification problems. Basic SVM is designed to solve binary classification problems and implements the structural risk minimization theory by seeking a maximum-margin

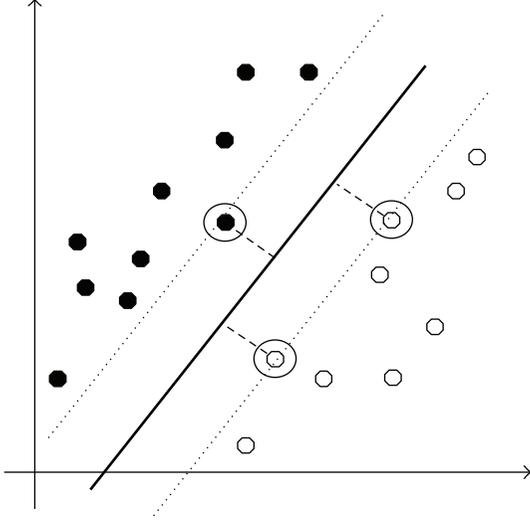


FIGURE 1: A linear SVM in 2-dimensional space with a maximum-margin hyperplane.

hyperplane between positive examples and negative examples in original space (linear SVM) or a high-dimensional feature space (nonlinear SVM with kernel trick). Figure 1 illustrates a linear SVM in two-dimensional space, where the examples on the boundary (two dashed lines with a maximum margin between two classes) are called support vectors, and the middle thick real line between these two dashed lines is the separator. An interesting result is that SVM can only use support vectors (a fraction of the set of all training examples) to form sparse solution and, thereby, has a fast classification speed.

Assume that there exists a training example set  $S = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$ , where  $x_i \in R^d$ ,  $y_i \in \{-1, +1\}$ , and  $y_i$  represents class labels of training examples  $x_i$ ,  $i = 1, 2, \dots, n$ . For seeking the maximum-margin hyperplane  $y = w^T \cdot x$ , the optimization objective of SVM is to minimize

$$F(w, \xi_i) = \frac{1}{2} (w^T \cdot w) + C \sum_i \xi_i, \quad (1)$$

subject to the constraints

$$y_i (w^T \cdot x_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0, \quad i = 1, 2, \dots, n, \quad (2)$$

where  $C$  is a penalty factor. To solve this quadratic optimization problem, we need to find the saddle point of the Lagrange function:

$$L_p(w, b, \alpha) = \frac{1}{2} w^T \cdot w + C \sum_{i=1}^n \xi_i - \sum_{i=1}^n \alpha_i [y_i (w^T \cdot x_i + b) - 1 + \xi_i] - \sum_{i=1}^n \beta_i \xi_i, \quad (3)$$

where  $\alpha_i, \beta_i$  are Lagrange multipliers and  $\alpha_i \geq 0, \beta_i \geq 0$ . By differentiating with respect to  $w$  and  $b$ , and using the Karush

Kuhn-Tucker (KKT) condition,  $L_p$  is transformed to the dual Lagrangian  $L_D(\alpha)$ :

$$\begin{aligned} \text{Max}_{\alpha} \quad L_D(\alpha) &= \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j y_i y_j (x_i^T \cdot x_j), \\ \text{subject to:} \quad &0 \leq \alpha_i \leq C, \\ &i = 1, 2, \dots, n, \\ &\sum_{i=1}^n \alpha_i y_i = 0. \end{aligned} \quad (4)$$

The solution  $\alpha^* = (\alpha_1^*, \alpha_2^*, \dots, \alpha_n^*)^T$  can be obtained by solving this quadratic optimization problem. Those  $\{x_i\}$  corresponding to nonzero  $\alpha_i^*$  are called support vectors. The parameters  $w^*$  and  $b^*$  of the optimal hyperplane can be obtained as follows:

$$w^* = \sum_{i=1}^{N_s} \alpha_i^* y_i x_i, \quad (5)$$

$$b^* = y_i - \sum_{i=1}^{N_s} \alpha_i^* y_i (x_i^T \cdot x_j),$$

where  $N_s$  denotes the total number of support vectors.

Then, the optimal hyperplane decision function  $f(x) = \text{sgn}((w^{*T} \cdot x) + b^*)$  can be written as

$$f(x) = \text{sgn} \left( \sum_{i=1}^{N_s} y_i \alpha_i^* (x_i^T \cdot x) + b^* \right). \quad (6)$$

SVM with linear kernel can be used to evaluate the weights of attributes. The decision function can be rewritten as

$$f(x) = \text{sgn} (w^{*T} \cdot x + b^*). \quad (7)$$

According to [43], the vector  $w^* = (w_1, w_2, \dots, w_d)^T$  can be used as weights of attributes, where  $d$  represents the total number of attributes. LSVM categorizes new data instances by testing whether the linear combination of the components of the vector  $x_i = (x_{i1}, x_{i2}, \dots, x_{id})^T$ ,  $w_1 x_{i1} + w_2 x_{i2} + \dots + w_d x_{id}$ , is above or below some threshold  $-b^*$ . It is easy to find that the larger the  $|w_j|$  the larger the impact of the corresponding  $j$ th attribute on the linear combination  $w_1 x_{i1} + w_2 x_{i2} + \dots + w_d x_{id}$ . This means that the final classification result is sensitive to the  $j$ th attribute, and then the  $j$ th attribute can be considered as being important for classification problem.

Attribute (or feature) evaluating using LSVM has a strict theoretical foundation [43]. Obviously, an attribute is considered as being important if it significantly influences the width of the margin between two classes. According to the theory of SVM, the margin is inversely proportional to the length  $\|w\|$  of  $w$ . For the solution  $w = \sum_i \alpha_i y_i x_i$  obtained by linear SVM (for the convenience of expression, the stars above  $w$  and  $\alpha_i$  are omitted),  $\|w\|^2$  can be regarded as a function of the

training vectors  $x_1, x_2, \dots, x_{N_s}$ , where  $x_i = (x_{i1}, x_{i2}, \dots, x_{id})$ , and thus the influence of feature  $j$  on  $\|w\|^2$  can be evaluated via absolute values of partial derivatives of  $\|w\|^2$  with respect to  $x_{ij}$ . This approach can provide an approximate analysis on importance extent of attributes although it neglects the fact that the values of the multipliers  $\alpha_i$  will change with training vectors changing [43].

For linear SVM, it turns out that

$$\sum_i \left| \frac{\partial \|w\|^2}{\partial x_{ij}} \right| = k |w_j|, \quad (8)$$

where the sum is over all support vectors and  $k$  is a constant independent of  $j$ . Thus, the attributes with the higher  $|w_j|$  are of the more important role in determining the width of the margin. Intuitively, this type of feature weighting seems to be appealing because features with small value of  $|w_j|$  do not have large influence on the output of SVM. Thus, these features can be considered as being unimportant for classification.

Support vector machine is based on the structural risk minimization theory and is an outstanding classification method. Meanwhile, the credit scoring problem is also a classification problem in essence. Hence, it is natural and reasonable that linear support vector machine is adopted to evaluate weights of attributes. In other words, the weights decided by LSVM are closely related to classification ability of classifiers.

**3.1.2. Evaluating Weights of Attributes via PCA.** As an alternative method, PCA is also used to evaluate weights of attributes in this paper. The main idea of PCA is to find the principal directions which best describe the distribution of credit samples within the entire credit sample space. The original variables are transformed to a set of new variables which are uncorrelated with each other and can be ranked from large to small in terms of variance such that the first several variables retain most of the variation in the entire original data.

Considering the set  $S' = \{x_1, x_2, \dots, x_n\}$ , which only contains input vectors of training examples in  $S = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$ . The vectors in  $S'$  are centered by  $c_i = x_i - a$ , where  $a = (1/n) \sum_{i=1}^n x_i$ , and then principal component analysis is performed to seek  $d$  orthonormal vectors,  $u_k$  ( $k = 1, 2, \dots, d$ ), which best describe the distribution of original data. The  $k$ th vector,  $u_k$ , is chosen such that

$$\lambda_k = \frac{1}{n} \sum_{i=1}^n (u_k^T c_i)^2 \quad (9)$$

is maximized, subject to

$$u_g^T u_k = \delta_{gk} = \begin{cases} 1, & \text{if } g = k, \\ 0, & \text{otherwise.} \end{cases} \quad (10)$$

Thus, the vectors  $u_k$  and the scalars  $\lambda_k$  are the eigenvectors and the eigenvalues, respectively, of the covariance matrix

$$C = \frac{1}{n} \sum_{i=1}^n c_i c_i^T = \frac{1}{n} A A^T, \quad (11)$$

where  $A = [c_1, c_2, \dots, c_n]$ .

The eigenvectors can be ranked via their corresponding eigenvalues from large to small to reflect their importance extent in characterizing the variation of original data. These eigenvectors span a new space, and all training samples are projected into the new space.

An example  $x$  is transformed to  $v = (v_1, v_2, \dots, v_h)$  in the new space by the following operation,

$$v_k = u_k^T (x - a), \quad k = 1, 2, \dots, h, \quad (12)$$

where  $h$  is the total number of the reserved eigenvectors.

The eigenvalue  $\lambda_k$  is the variance of original data in the direction of the eigenvector  $u_k$  and hence reflects the capability of the  $k$ th attribute in the new space in describing original data. As such,  $\alpha_k = \lambda_k / \sum_{i=1}^h \lambda_i$  can be defined as the importance percentage of the  $k$ th attribute in the new space.

PCA and LSVM provide two different ways of evaluating weights of attributes. LSVM method selects attributes in the original space while PCA method selects attributes in the transformed feature space. Additionally, weights of attributes decided by LSVM reflect the classification ability of the corresponding attributes, and those obtained by PCA reflect the description capability for original data distribution.

**3.2. Weight-Selected Attribute Bagging.** After obtaining weights of attributes, we can select attributes based on probabilities decided by weights of attributes and train multiple classifiers using different attribute subsets, respectively, to perform attribute bagging.

The essence of weight-selected attribute bagging (WSAB) lies in the way of selecting attributes for each individual classifier in an ensemble (bagging). Concretely speaking, weights of attributes are used to construct many different attribute subsets such that the attributes with the larger weights have the larger probabilities to be selected into each attribute subset. The selection of attributes for each single attribute subset does not permit repetition of attributes that is, there are no repeated attributes in each attribute subset, but the same attributes can be chosen into different attribute subsets. Thus, the subsets containing unimportant features only can be avoided with a larger probability, compared to randomly selected attribute bagging (RSAB), so that the classification accuracies of individual classifiers can be guaranteed. On the other hand, the diversity of individual classifiers can be still ensured because attributes are chosen by probabilities and there exist differences among different attribute subsets.

Subsequently, like standard attribute bagging (AB), projections of training examples onto these attribute subsets are created. Individual scoring models are trained based on each projection, respectively, and all individual scoring models are aggregated by a specific strategy for test instances.

The appropriate size of attribute subset can be determined by cross-validation technique. The original training set is divided into two parts—a new training set and a validation set. With the attribute subset size changing from 1 to  $d$  (weights are decided by linear SVM) or  $h$  (weights are decided by PCA),  $d$  (or  $h$ ) WSAB models with different attribute subset sizes are created, respectively. Then, the validation set is used to test these WSAB models to find out the optimal attribute subset size.

The main steps of WSAB are as follows.

*Step 1.* Compute weights of attributes,  $w_1, w_2, \dots, w_d$  (or  $w_h$ ), via attribute evaluating method.

*Step 2.* Decide an appropriate attribute subset size,  $m$ , by cross-validation.

*Step 3.* Generate a series of attribute subsets through repeating the following substeps.

*Substep 1.* Construct an array with  $l$  elements ( $l$  is large enough). The  $i$ th attribute takes  $w_i / \sum_j w_j$  part of the array,  $i = 1, 2, \dots, d$  or  $h$ .

*Substep 2.* Perform the following cycle to construct an attribute subset with  $m$  attributes:

- (A)  $t = 0$ ;
- (B) randomly select an element of the array into the subset;
- (C) delete all positions of the chosen element from the array;
- (D)  $t = t + 1$ ;
- (E) if  $t = m$ , one attribute subset is created; else go to (B).

*Step 4.* Create projections of training examples onto the selected attribute subsets.

*Step 5.* Train individual classification models based on each projection, respectively, and use all individual scoring models to vote for test instances.

The modeling process of WSAB is illustrated in Figure 2.

## 4. Experimental Results and Comparisons

*4.1. Credit Datasets.* Two datasets are described in Table 1, which are German credit dataset and Australian credit dataset from the UCI Machine Learning Repository [39]. In German credit dataset, there are 1000 instances which contain 700 instances of creditworthy applicants and 300 instances regarded as bad credit applicants. Each example consists of 24 predictive attributes. Australian credit dataset includes 690 observers which record 307 good credit applicants and 383 bad credit applicants, and every instance consists of 14 predictive attributes.

*4.2. Experimental Settings.* We randomly divided the whole dataset into two parts—training set and test set. Training set

takes two-thirds of the whole dataset and testing set takes one third of the whole dataset. The SVM with a Gaussian kernel was chosen as the basic classifier. Grid search in training set was used to decide the best parameters of SVM. The PR\_tools [44] developed by MATLAB language was utilized as experimental platform. Each bagging model was repeated for 30 times, and then the average accuracy and the average standard deviation of accuracy were computed.

### 4.3. Evaluating Weights of Attributes

*4.3.1. Evaluating Weights of Attributes via LSVM.* The LSVM was performed on training dataset to obtain the weight of the  $j$ th attribute,  $w_j$ . Then the importance percentage of the  $j$ th attribute was computed as  $w_j / \sum_i w_i$ . The results are shown in Figure 3 for Australian dataset and Figure 4 for German dataset, respectively.

For Australian dataset, the top four important attributes make up 61.30% of the sum of all attribute weights, while the four most unimportant attributes only make up 3.82%. In addition, the most important one makes up 28.91% of the sum of all attribute weights and the most unimportant one makes up only 0.37% of the sum. The statistical results mean that, for Australian dataset, a few attributes are very important yet some other attributes contribute less to classification.

For German dataset, the top seven important attributes make up 56.33% of the sum of all attribute weights, while the four most unimportant attributes only make up 8.46%. Meanwhile, the most important attribute makes up 12.76% of the sum of all attribute weights and the most unimportant one makes up only 0.71% of the sum. Compared to Australian dataset, we can conclude that the importance extent of each attribute in German dataset is more evenly distributed.

*4.3.2. Evaluating Weights of Attributes via PCA.* We also performed PCA to calculate weights of attributes. Just as described in Section 3, weights of attributes obtained by linear SVM reflect the classification ability of attributes, but those obtained by PCA mainly describe the distribution of the original data.

The eigenvalues  $\lambda_i$  and their corresponding eigenvectors of the covariance matrix for each dataset were calculated. Then, all training instances were projected onto the space spanned by  $h$  orthonormal eigenvectors. The importance percentage of each new attribute can be presented as  $\lambda_j / \sum_{i=1}^h \lambda_i$ . Analogous to the analysis on weights of attributes from LSVM, the distribution of the attribute importance obtained by PCA is shown in Figure 5 for Australian dataset and Figure 6 for German dataset.

For Australian dataset, the top four important attributes make up 78.3% of the sum of all attribute weights, while the four most unimportant attributes only make up 1.80%. Additionally, the most important attribute makes up 30.2% of the sum of all attribute weights and the most unimportant one makes up only 0.23%. The results reflect the fact that several main attributes contribute more to the description of the original data for Australian dataset, yet some other attributes provide less information for the original data.

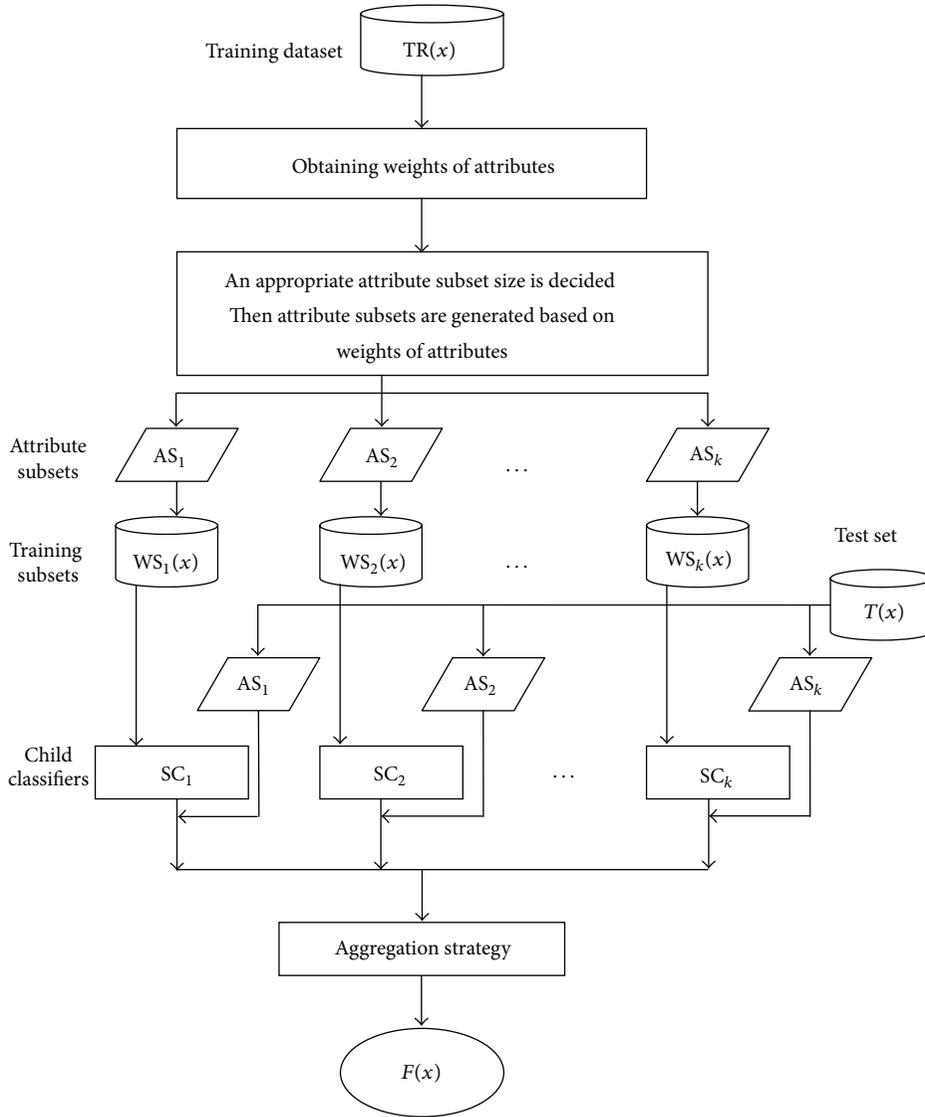


FIGURE 2: Weight-selected attribute bagging model.

TABLE 1: Description of credit datasets.

	German credit	Australian credit
Number of observers	1000	690
Number of predictive attributes	24	14
Percentage of good credit	70%	44.5%
Percentage of bad credit	30%	55.5%

For German dataset, the top seven important attributes make up 63.25% of the sum of all attribute weights, while the seven most unimportant attributes only make up 5.85%. The most important attribute makes up 13.5% of the sum of all attribute weights and the most unimportant attribute makes up only 0.33%. Therefore, the importance of each attribute for German dataset is more evenly distributed, compared to Australian dataset.

Interestingly, the results obtained via PCA are similar to those via LSVM not only for Australian dataset but also for German dataset, although LSVM and PCA adopt different approaches to calculating the weights of attributes one for data classification, and the other for data description.

#### 4.4. Performance Comparison on Different Attribute Bagging Methods with Different Attribute Subset Sizes

4.4.1. Comparison on Accuracy. The size of attribute subset is critical for attribute bagging. Hence, this section will evaluate classification accuracy of the WSAB with different sizes of attribute subsets. Meanwhile, several other related methods were also compared.

For the convenience of expression, the WSAB using LSVM to determine weights of attributes is abbreviated as LSVM-WSAB; the WSAB using PCA to calculate weights of

attributes is denoted as PCA-WSAB; the randomly selected attribute bagging is written as RSAB. For each bagging method as well as for each size of attribute subset ranging from 1 to  $d$  (or  $h$ ), 45 classifiers were built to perform voting for test instances. Concretely speaking, for each attribute bagging method, 45 attribute subsets with the same subset size  $m$  were created, and then training examples and test examples were, respectively, projected onto the  $m$  selected attributes. Subsequently, 45 SVMs corresponding to 45 attribute subsets, respectively, were trained to vote for test instances. For each attribute subset size and each bagging method, 30 trials of the above process were performed and their results were averaged to evaluate classification accuracy. Furthermore, in order to prove the superiority of the voting, each attribute bagging method was also compared with the best single classifier. The best single classifier is represented as BS-SVM. The final results are given in Figure 7 for Australian dataset and Figure 8 for German dataset.

From Figures 7 and 8, the accuracy of WSAB is not high for small size of attribute subset, but rises gradually as the size of attribute subset becomes larger; then the accuracy tends to decrease when the size of attribute subset becomes large enough. Meanwhile, RSAB has the same behavior as WSAB. However, when the size of attribute subset is small, LSVM-WSAB and PCA-WSAB are more accurate than RSAB. In addition, LSVM-WSAB and PCA-WSAB are able to use smaller size of attribute subset to reach the maximum accuracy than RSAB. Meanwhile, the maximum accuracies of LSVM-WSAB and PCA-WSAB are higher than that of RSAB. We can provide a reasonable explanation for this phenomenon. For too small attribute subsets, individual classifiers used for voting have low accuracies since the information used for classification is lost too much. On the other side, for too large attribute subsets, the diversity among all members in ensemble decreases, so that the ensemble effect is affected. Additionally, for RSAB, the attributes in attribute subsets are selected randomly. Hence, RSAB needs larger sizes of attribute subsets to acquire sufficient information in order to reach its highest classification accuracy. However, WSAB selects attributes in terms of weights, such that WSAB can use smaller sizes of attribute subsets to contain most of important attributes and then to achieve its highest accuracy.

LSVM-WSAB, PCA-WSAB and RSAB are more accurate than BS-SVM for large size of attribute subset. The maximum accuracies of LSVM-WSAB, PCA-WSAB, and RSAB are higher than that of BS-SVM. The results prove that attribute bagging can improve effectively the performance of single classifier.

Additionally, PCA-WSAB needs less attributes to reach the maximum accuracy than LSVM-WSAB. The reason lies in the fact that PCA eliminates the correlation among attributes and the important attributes are more concentrated on several eigenvectors.

Moreover, an interesting finding for WSAB is that small size of attribute subsets can achieve high accuracy for Australian dataset, whereas for German dataset, accuracy of WSAB rises slowly with the size of attribute subset increasing. As we mentioned before, the important attributes in Australian dataset are concentrated on only several variables,

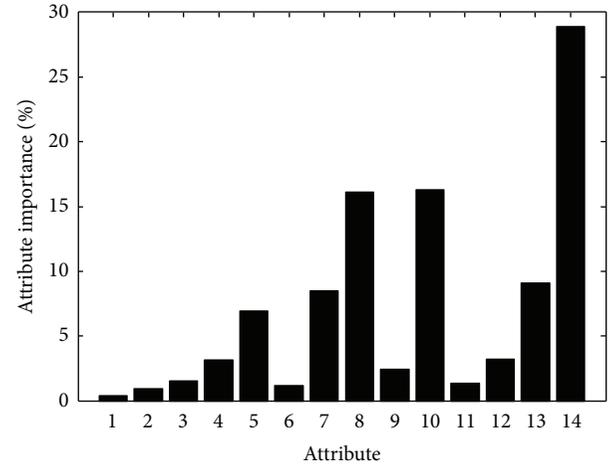


FIGURE 3: Importance percentage of each attribute for Australian dataset.

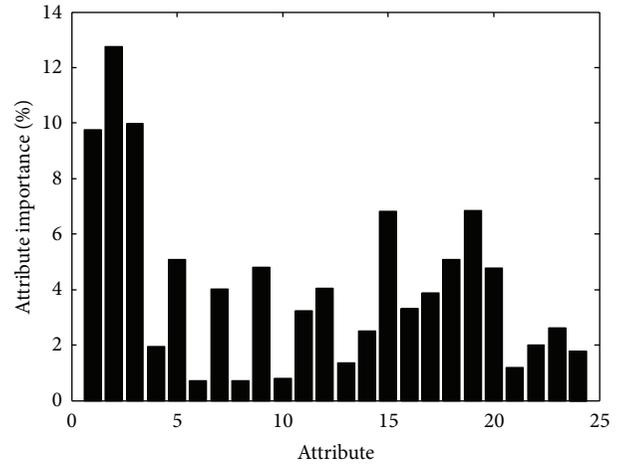


FIGURE 4: Importance percentage of each attribute for German dataset.

and the remaining attributes contribute less to classification. Hence, WSAB model can acquire enough information using a small size of attribute subset for Australian dataset. However, for German dataset, the importance of attributes is more evenly distributed, and thus WSAB needs larger size of attribute subset to obtain enough information for classification.

**4.4.2. Comparison on Stability.** When computing average accuracy of 30 trials for each attribute bagging model as well as for each attribute subset size, the standard deviation of accuracy was also computed to evaluate the classification stability of attribute bagging models. The standard deviations are shown in Figure 9 for Australian dataset and Figure 10 for German dataset.

From Figures 9 and 10, when the size of attribute subset is small, the standard deviation of accuracy of WSAB is larger than RSAB; when the size of attribute subset becomes larger gradually, the standard deviation of accuracy of WSAB

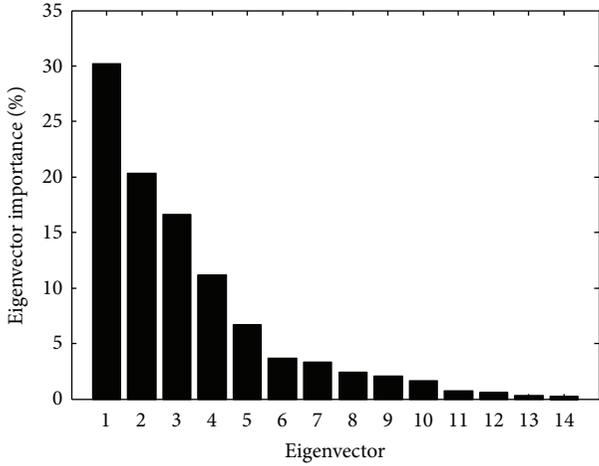


FIGURE 5: Importance percentage of each new attribute for Australian dataset.

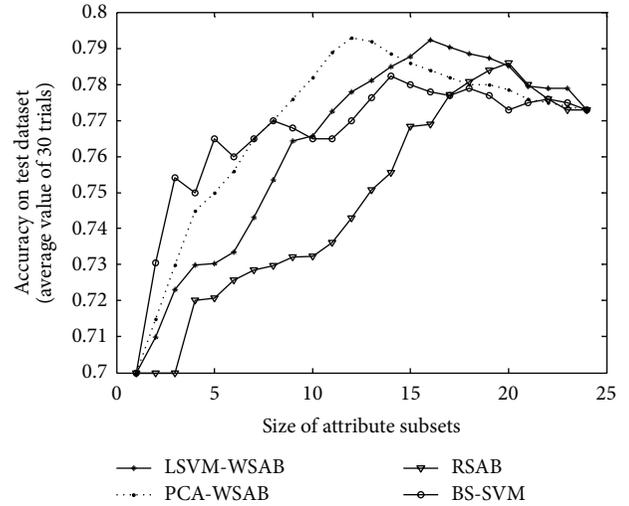


FIGURE 8: Average accuracies of bagging methods and BS-SVM with the size of attribute subset changing based on German dataset.

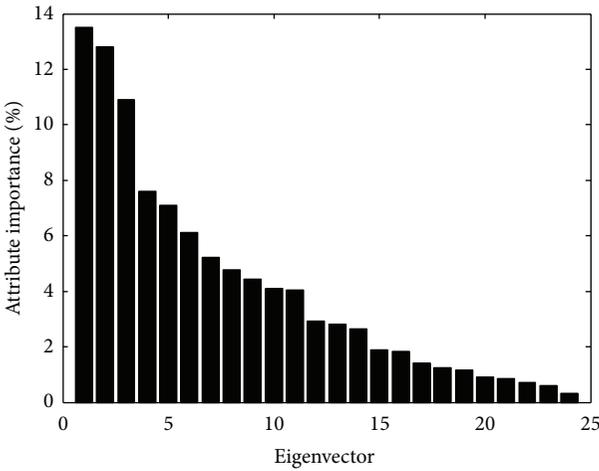


FIGURE 6: Importance percentage of each new attribute for German dataset.

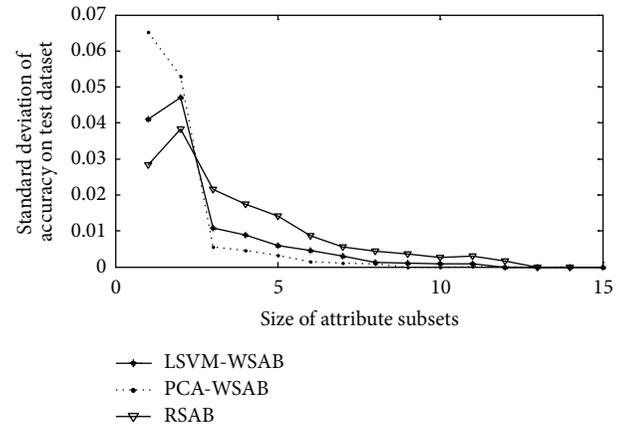


FIGURE 9: Standard deviations for three attribute bagging methods: LSVM-WSAB, PCA-WSAB, and RSAB based on Australian dataset.

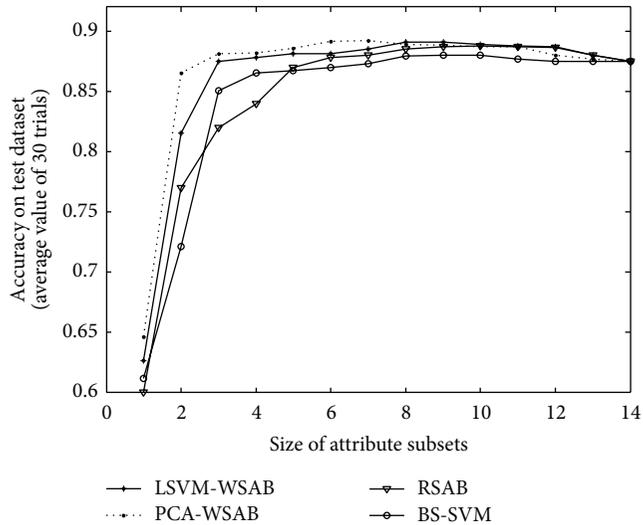


FIGURE 7: Average accuracies of bagging methods and BS-SVM with the size of attribute subset changing based on Australian dataset.

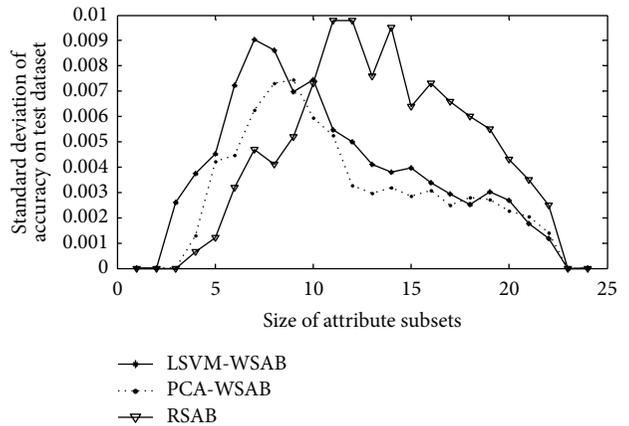


FIGURE 10: Standard deviations for three attribute bagging methods: LSVM-WSAB, PCA-WSAB, and RSAB based on German dataset.

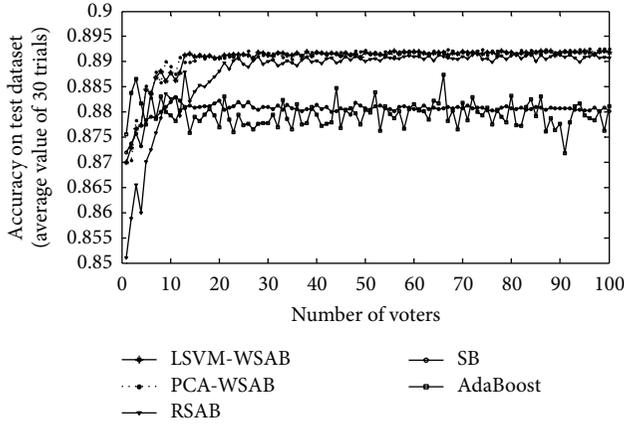


FIGURE 11: The average accuracies of attribute bagging, standard bagging (SB), and AdaBoost with the number of voters changing for Australian dataset.

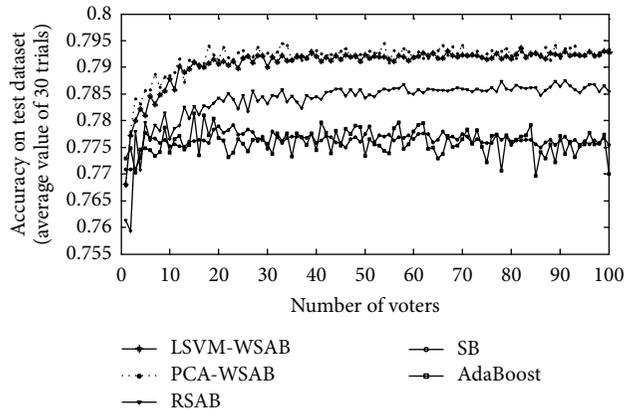


FIGURE 12: The average accuracies of attribute bagging, standard bagging (SB), and AdaBoost with the number of voters changing for German dataset.

become smaller than RSAB. This is because when the size of attribute subset is small, most of attribute subsets may not contain important attributes, so the accuracy of RSAB is always low. Meanwhile, for small attribute subsets, some of them for WSAB may contain important attributes, yet the other some may not contain any important attributes; so the performance of WSAB model is not stable. But for large attribute subsets, most of important attributes can be chosen into attribute subsets with large probability, resulting in the stable classification performance of WSAB. On the other hand, RSAB randomly chooses attributes into attribute subsets, each attribute with the same probability; so larger difference exists between different attribute subsets for RSAB than for WSAB. Therefore, the performance of RSAB becomes less stable than WSAB when the size of attribute subset becomes larger. Furthermore, when WSAB and RSAB adopt their optimal sizes of attribute subsets, respectively, the standard deviation of accuracy of WSAB is smaller than that of RSAB.

The highest accuracy of each model and the corresponding standard deviation (Std) are shown in Table 2 for Australian dataset and Table 3 for German dataset.

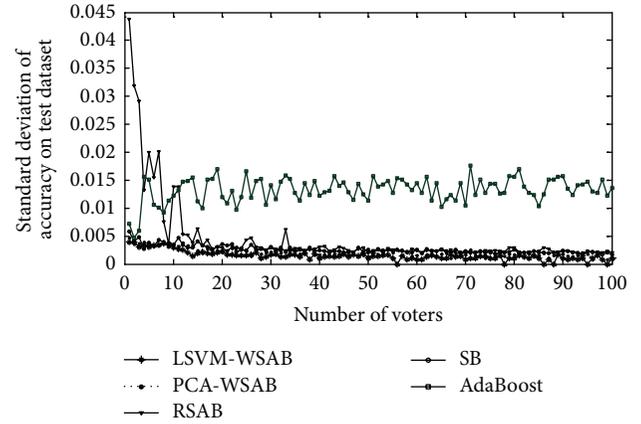


FIGURE 13: Standard deviations of accuracy of attribute bagging, standard bagging, and AdaBoost with the number of voters changing for Australian dataset.

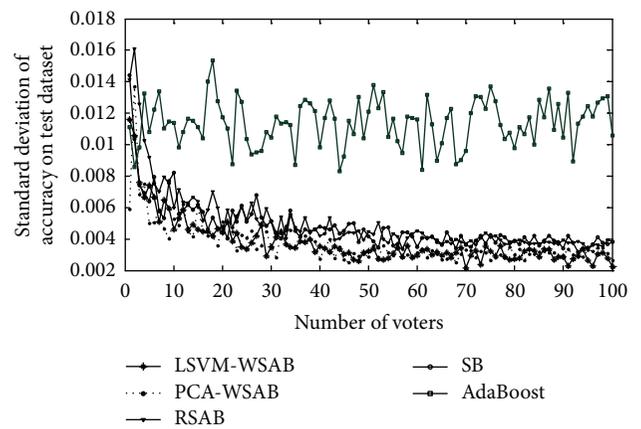


FIGURE 14: Standard deviations of accuracy of attribute bagging, standard bagging, and AdaBoost with the number of voters changing for German dataset.

From Tables 2 and 3, WSAB model performs better than RSAB both in accuracy and stability, and all attribute bagging methods have higher accuracies than BS-SVM.

#### 4.5. Performance Comparison on Different Ensemble Methods with Different Numbers of Voters

**4.5.1. Comparison on Accuracy.** In this section, each attribute bagging model adopts its optimal attribute subset size, which is calculated by cross-validation. Then, we compare the accuracies of attribute bagging models including LSVM-WSAB, PCA-WSAB, and RSAB, as well as data partitioning ensemble models including standard bagging (SB) and AdaBoost, with the number of voters changing. For each number of voters and for each model, the experiments were also repeated 30 times, and the average accuracy was computed. The results are illustrated in Figure 11 for Australian dataset and Figure 12 for German dataset.

From Figures 11 and 12, WSAB has higher accuracy than SB and AdaBoost for almost each number of voters.

TABLE 2: The highest accuracy of each model and the corresponding standard deviation (Std) for Australian dataset.

	LSVM-WSAB (8 attributes)	PCA-WSAB (6 attributes)	RSAB (10 attributes)	BS-SVM (9 attributes)	SVM (14 attributes)
Accuracy	0.8912	0.8918	0.8880	0.8800	0.8750
Std	0.0013	0.0015	0.0030	0	0

TABLE 3: The highest accuracy of each model and the corresponding standard deviation (Std) for German dataset.

	LSVM-WSAB (16 attributes)	PCA-WSAB (12 attributes)	RSAB (20 attributes)	BS-SVM (14 attributes)	SVM (24 attributes)
Accuracy	0.7924	0.7930	0.7860	0.7824	0.7729
Std	0.0034	0.0033	0.0043	0	0

This is because WSAB can reserve important attributes and eliminate some redundant and uninformative attributes by large probability. When the number of voters is small, SB and AdaBoost have higher accuracies than RSAB. But when the number of voters is more than 10, RSAB has higher accuracy than standard bagging and AdaBoost. The reason is that RSAB needs more voters in order to include most of important attributes.

Moreover, the accuracy of standard bagging model increases gradually before the number of voters reaches 20, and its accuracy maintains at a certain level after the number of the voters is larger than 20. The accuracy of AdaBoost fluctuates most sharply with the number of voters changing. Meanwhile, the accuracies of attribute bagging models increase quickly before the number of voters reaches 20 and then their accuracies also maintain at certain levels. This is because standard bagging and AdaBoost sample from training dataset with all attributes for each voter, whereas attribute bagging only uses part of attributes. Therefore, attribute bagging needs to use more voters to “cover” all attributes, and with the number of voters increasing, more information is integrated into bagging model. The higher accuracies achieved by attribute bagging models support the conclusion that attribute bagging models are superior to data partitioning ensemble models.

For small number of voters, both LSVM-WSAB and PCA-WSAB perform better than RSAB. This further proves our idea that WSAB model can utilize important attributes to obtain better classification results. For large numbers of voters, WSAB model performs slightly better than RSAB for Australian dataset and much better than RSAB for German dataset. Therefore, the conclusion can be made that WSAB outperforms RSAB.

*4.5.2. Comparison on Stability.* Besides computing average accuracies of 30 trials for each number of voters, we also computed the standard deviation of accuracy to evaluate the classification stability of different methods. The standard deviations of classification accuracy for each model are shown in Figure 13 for Australian dataset and Figure 14 for German dataset.

From Figures 13 and 14, for small number of voters, WSAB has almost the same standard deviation as SB, and

RSAB has higher standard deviation than SB. But for large number of voters, WSAB has lower standard deviation than SB, and SB has nearly the same standard deviation as SB. Additionally, when the number of voters is more than 10, the standard deviation of accuracy of AdaBoost is much larger than other methods. This supports the conclusion that boosting is easy to bias on several samples which are difficult to be classified. Therefore, boosting is not stable for the credit scoring problem; sometimes effective and sometimes not.

Furthermore, WSAB is more stable than RSAB. The reason is that WSAB can select important attributes for each child classifier, such that the accuracies of child classifiers in WSAB fluctuate less than those of RSAB. Therefore, from the viewpoint of the whole results, WSAB is more stable than RSAB.

When the number of voters is larger than 50, the accuracy and stability of each ensemble model maintain certain levels. Therefore, in order to compare the performance of all ensemble models, we show the accuracies of all ensemble models in Table 4 for Australian dataset and Table 5 for German dataset when the number of voters is 50.

From Tables 4 and 5, WSAB model performs best both on accuracy and stability, and RSAB model follows.

## 5. Conclusions and Future Research

This paper presents the WSAB for credit risk evaluation. The implementation of WSAB includes two steps. The first step is to determine weights of attributes. During the second step, attributes are selected into attribute subsets according to the probabilities determined by attribute weights. This method of modeling makes the WSAB have two advantages, namely, improving the accuracy of each individual classifier in ensemble and increasing the diversity among all individual classifiers. For the first merit, the more important attributes can be incorporated into each attribute subset with the larger probabilities so that each individual classifier can acquire high classification accuracy. For the second merit, the way of selecting attributes by probability makes different attribute subsets have different unimportant attributes which are of small weights, and consequently the diversity among different classifiers can be still guaranteed. In fact, accuracy and diversity are two critical factors for bagging. Experimental

TABLE 4: The accuracy of each model and the corresponding standard deviation for Australian dataset.

	LSVM-WSAB	PCA-WSAB	RSAB	SB	AdaBoost
Accuracy	0.8919	0.8922	0.8890	0.8807	0.8779
Std	0.0011	0.0013	0.0020	0.0026	0.0114

TABLE 5: The accuracy of each model and the corresponding standard deviation for German dataset.

	LSVM-WSAB	PCA-WSAB	RSAB	SB	AdaBoost
Accuracy	0.7918	0.7924	0.7842	0.7775	0.7731
Std	0.0032	0.0029	0.0045	0.0041	0.0121

results also confirm the superiority of WSAB over randomly selected attribute bagging (RSAB), especially over standard bagging, AdaBoost, and individual classifier.

Broadly speaking, the WSAB provides a framework of evaluating credit risk. In this framework, any attribute weighting method and any basis classifier can be combined. This paper adopts two completely different ways to compute weights of attributes: LSVM and PCA. The weights obtained by LSVM emphasize the classification ability of attributes, and the weights from PCA reflect the description ability of attributes for original data. However, credit scoring is just considered as a classification problem, for which LSVM seems to be more suitable than PCA, and experimental results also demonstrate the conclusion.

The next work will attempt to combine other approaches of computing weights of attributes and other basis classifiers to perform credit risk evaluation and then to compare their performances in terms of accuracy and stability. Additionally, the WSAB can also be applied to other practical systems, such as stock market prediction [45] and MRI brain image classification [46].

## Acknowledgments

The authors would like to thank the anonymous reviewers for their constructive comments and suggestions which have led to great improvement on this paper. This work is supported by the National Natural Science Foundation of China (no. 61271374) and the Beijing Natural Science Foundation (no. 4122068).

## References

- [1] H. L. Chen, B. Yang, G. Wang et al., "A novel bankruptcy prediction model based on an adaptive fuzzy  $k$ -nearest neighbor method," *Knowledge-Based Systems*, vol. 24, no. 8, pp. 1348–1359, 2011.
- [2] B. Yang, L. X. Li, Q. Xie, and J. Xu, "Development of a KBS for managing bank loan risk," *Knowledge-Based Systems*, vol. 14, no. 5-6, pp. 299–302, 2001.
- [3] L. C. Thomas, D. B. Edelman, and J. N. Crook, *Credit Scoring and Its Applications*, SIAM Monographs on Mathematical Modeling and Computation, SIAM, Philadelphia, Pa, USA, 2002.
- [4] J. N. Crook, D. B. Edelman, and L. C. Thomas, "Recent developments in consumer credit risk assessment," *European Journal of Operational Research*, vol. 183, no. 3, pp. 1447–1465, 2007.
- [5] J. H. Myers and E. W. Forgy, "The development of numerical credit evaluation systems," *Journal of the American Statistical Association*, vol. 58, no. 303, pp. 799–806, 1963.
- [6] J. C. Wiginton, "A note on the comparison of logit and discriminant models of consumer credit behavior," *Journal of Financial Quantitative Analysis*, vol. 15, no. 3, pp. 757–770, 1980.
- [7] P. Makowski, "Credit scoring branches out," *Credit World*, vol. 74, no. 2, pp. 30–37, 1985.
- [8] X. Y. Zhou, D. F. Zhang, and Y. Jiang, "A new credit scoring method based on rough sets and decision tree," in *Advances in Knowledge Discovery and Data Mining*, vol. 5012 of *Lecture Notes in Computer Science*, pp. 1081–1089, Springer, New York, NY, USA, 2008.
- [9] W. E. Henley and D. J. Hand, "Construction of a  $k$ -nearest-neighbor credit-scoring system," *IMA Journal of Management Mathematics*, vol. 8, no. 4, pp. 305–321, 1997.
- [10] H. L. Jensen, "Using neural networks for credit scoring," *Managerial Finance*, vol. 18, no. 6, pp. 15–26, 1992.
- [11] D. West, "Neural network credit scoring models," *Computers and Operations Research*, vol. 27, no. 11-12, pp. 1131–1152, 2000.
- [12] V. S. Desai, D. G. Conway, J. N. Crook, and G. A. J. R. Overstreet, "Credit-scoring models in the credit-union environment using neural networks and genetic algorithms," *IMA Journal of Management Mathematics*, vol. 8, no. 4, pp. 323–346, 1997.
- [13] M. B. Yobas, J. N. Crook, and P. Ross, "Credit scoring using neural and evolutionary techniques," *IMA Journal of Mathematics Applied in Business and Industry*, vol. 11, no. 2, pp. 111–125, 2000.
- [14] D. F. Zhang, H. Y. Huang, Q. S. Chen, and Y. Jiang, "A comparison study of credit scoring models," in *Proceedings of the 3rd International Conference on Natural Computation (ICNC '07)*, vol. 1, pp. 15–18, Haikou, China, August 2007.
- [15] H. A. Abdou, "Genetic programming for credit scoring: the case of Egyptian public sector banks," *Expert Systems with Applications*, vol. 36, no. 9, pp. 11402–11417, 2009.
- [16] J. J. Huang, G. H. Tzeng, and C. S. Ong, "Two-stage genetic programming (2SGP) for the credit scoring model," *Applied Mathematics and Computation*, vol. 174, no. 2, pp. 1039–1053, 2006.
- [17] K. Leung, F. Cheong, and C. Cheong, "Consumer credit scoring using an artificial immune system algorithm," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '07)*, pp. 3377–3384, Singapore, September 2007.
- [18] K. K. Lai, L. Yu, L. G. Zhou, and S. Y. Wang, "Credit risk evaluation with least square support vector machine," in *Rough Sets and Knowledge Technology*, vol. 4062 of *Lecture Notes in*

- Computer Science*, pp. 490–495, Springer, New York, NY, USA, 2006.
- [19] K. B. Schebesch and R. Sleeking, “Support vector machines for classifying and describing credit applicants: detecting typical and critical regions,” *Journal of the Operational Research Society*, vol. 56, no. 9, pp. 1082–1088, 2005.
- [20] D. Zhang, X. Y. Zhou, S. C. H. Leung, and J. Zheng, “Vertical bagging decision trees model for credit scoring,” *Expert Systems with Applications*, vol. 37, no. 12, pp. 7838–7843, 2010.
- [21] S. L. Lin, “A new two-stage hybrid approach of credit risk in banking industry,” *Expert Systems with Applications*, vol. 36, no. 4, pp. 8333–8341, 2009.
- [22] N. C. Hsieh, “Hybrid mining approach in the design of credit scoring models,” *Expert Systems with Applications*, vol. 28, no. 4, pp. 655–665, 2005.
- [23] D. Zhang, M. Hifi, Q. Chen, and W. Ye, “A hybrid credit scoring model based on genetic programming and support vector machines,” in *Proceedings of the 4th International Conference on Natural Computation (ICNC '08)*, vol. 7, pp. 8–12, Jinan, China, October 2008.
- [24] L. Zhou, K. K. Lai, and L. Yu, “Least squares support vector machines ensemble models for credit scoring,” *Expert Systems with Applications*, vol. 37, no. 1, pp. 127–133, 2010.
- [25] T. G. Dietterich, “Machine-learning research: four current directions,” *AI Magazine*, vol. 18, no. 4, pp. 97–136, 1997.
- [26] R. O. Duda, P. H. Hart, and D. G. Stork, *Pattern Classification*, Wiley-Interscience, New York, NY, USA, 2000.
- [27] L. Breiman, “Bagging predictors,” *Machine Learning*, vol. 24, no. 2, pp. 123–140, 1996.
- [28] L. Breiman, “Bias, variance, and arcing classifiers,” Tech. Rep. 460, University of California, Department of Statistics, Berkeley, Calif, USA, 1996.
- [29] Y. Freund and R. Schapire, “Experiments with a new boosting algorithm,” in *Proceedings of the 13th International Conference on Machine Learning*, pp. 148–156, Bari, Italy, 1996.
- [30] R. Bryll, R. Gutierrez-Osuna, and F. Quek, “Attribute bagging: improving accuracy of classifier ensembles by using random feature subsets,” *Pattern Recognition*, vol. 36, no. 6, pp. 1291–1302, 2003.
- [31] K. Tumer and N. C. Oza, “Decimated input ensembles for improved generalization,” in *Proceedings of the International Joint Conference on Neural Networks (IJCNN '99)*, pp. 3069–3074, Washington, DC, USA, July 1999.
- [32] K. Tumer and J. Ghosh, “Classifier combining: analytical results and implications, Working notes from the workshop ‘integrating multiple learned models,’” in *Proceedings of the 13th National Conference on Artificial Intelligence*, Portland, Ore, USA, August 1996/1996.
- [33] D. Opitz, “Feature selection for ensembles,” in *Proceedings of the 16th AAAI National Conference on Artificial Intelligence*, pp. 379–384, Orlando, Fla, USA, 1999.
- [34] D. Opitz and R. Maclin, “Popular ensemble methods: an empirical study,” *Journal of Artificial Intelligence Research*, vol. 11, pp. 169–198, 1999.
- [35] L. Nanni and A. Lumini, “An experimental comparison of ensemble of classifiers for bankruptcy prediction and credit scoring,” *Expert Systems with Applications*, vol. 36, no. 2, pp. 3028–3033, 2009.
- [36] G. Wang, J. Ma, L. Huang, and K. Xu, “Two credit scoring models based on dual strategy ensemble trees,” *Knowledge-Based Systems*, vol. 26, pp. 61–68, 2012.
- [37] P. Ravisankar and V. Ravi, “Financial distress prediction in banks using group method of data handling neural network, counter propagation neural network and fuzzy ARTMAP,” *Knowledge-Based Systems*, vol. 23, no. 8, pp. 823–831, 2010.
- [38] C. F. Tsai, “Feature selection in bankruptcy prediction,” *Knowledge-Based Systems*, vol. 22, no. 2, pp. 120–127, 2009.
- [39] A. Asuncion and D. J. Newman, *UCI Machine Learning Repository*, School of Information and Computer Science, University of California, Irvine, Calif, USA, 2007, <http://www.ics.uci.edu/~mllearn/MLRepository.html>.
- [40] Y. Freund and R. E. Schapire, “A decision-theoretic generalization of on-line learning and an application to boosting,” *Journal of Computer and System Sciences*, vol. 55, no. 1, pp. 119–139, 1997.
- [41] C. Guerra-Salcedo and D. Whitley, “Genetic approach to feature selection for ensemble creation,” in *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO '99)*, pp. 236–243, Morgan Kaufmann, 1999.
- [42] V. Vapnik, *The Nature of Statistical Learning Theory*, Springer, Berlin, Germany, 1995.
- [43] B. Janez, G. Marko, M. Natasa, and M. Dunja, “Feature selection using linear support vector machines,” Tech. Rep. MSR-TR-2002-63, Microsoft Research Microsoft Corporation, 2002.
- [44] R. P. W. Duin, P. Juszczak, P. Paclik et al., PRTools4.1, A Matlab Toolbox for Pattern Recognition, 2007, <http://www.prtools.org/>.
- [45] Y. Zhang and L. Wu, “Stock market prediction of S&P 500 via combination of improved BCO approach and BP neural network,” *Expert Systems with Applications*, vol. 36, no. 5, pp. 8849–8854, 2009.
- [46] Y. Zhang, Z. Dong, L. Wu, and S. Wang, “A hybrid method for MRI brain image classification,” *Expert Systems with Applications*, vol. 38, no. 8, pp. 10049–10053, 2011.

## Research Article

# Design of Special Impacting Filter for Multicarrier ABPSK System

Zhimin Chen and Lenan Wu

*School of Information Science & Engineering, Southeast University, 2 Sipailou, Nanjing 210096, China*

Correspondence should be addressed to Zhimin Chen; [chenzhimin\\_118@163.com](mailto:chenzhimin_118@163.com)

Received 14 January 2013; Accepted 17 March 2013

Academic Editor: Yudong Zhang

Copyright © 2013 Z. Chen and L. Wu. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A rather intuitive technique known as pole-zero placement is introduced to illustrate the frequency response of the special impacting filters (SIFs) with a pair of conjugate zero-poles and deduce the equation of the pole radii. Based on that, the paper proposes an iterative scheme to derive the parameters of the cascade notch filter. The cost function is determined by the cascading notch filter's influence on impacting filters, converting the cost function's least square problem to a filter parameters' standard quadratic programming problem. Finally, a cascading notch SIF (CNSIF) designed to demodulate the ABPSK signals is realized.

## 1. Introduction

As we know, spectrum is an unrenovable resource, and the new wireless communication systems are occupying more and more transmission bandwidth. So many corporations and research institutions try to find the optimal method to utilize the spectrum. Against this background, the ultranarrow band (UNB) technology was proposed by doctor Walker. Up until now, various modulation methods have been successively proposed with the progress of UNB development, for example, variable phase shifting keying (VPSK), enhanced VPSK, very minimum shifting keying (VMSK), pulse position phase reversal keying (3PRK), missing cycle modulation (MCM), suppressed cycle modulation (SCM), and minimum sideband modulation (MSB), and so forth. The key of the demodulation is the "zero group delay" filter. The "zero group delay" means the rise time of the filter is very short and the phase changes are preserved after the "zero group delay" filter. The core unit is a quartz crystal which is too unreliable, instable, and inflexible to be mass-produced [1–3]. Wu et al. analyzed the feasibility of the UNB and proposed a high-efficiency modulation called extended BPSK (EBPSK) [4, 5]. In 2009, they proposed continuous phase EBPSK (CP-EBPSK) based on the EBPSK [6], further tightening and reducing the EPBSK power spectrum

sideband. Subsequently, they imported the pseudorandom sequence and modified spectrum parameter to improve the CP-EBPSK, that is, modified CP-EBPSK (MCP-EBPSK). All of the above techniques have the same characteristic, that is "asymmetric." They are uniformly referred to as asymmetric binary phase shift keying (ABPSK). ABPSK could increase the transport bitrate and the spectrum utilization efficiency with a more narrow band [7]. The modulation of the ABPSK signal relies on the special impacting filter (SIF), which could convert phase changes to amplitude impacts. Details of the SIF will be given in Section 3.

In recent years, multicarrier (MC) scheme has been recognized as a potential candidate for the physical layer of the wireless communication systems which could not only gain spectral efficiency but also lower out-of-band radiation. The first candidate of MC schemes is orthogonal frequency-division multiplexing (OFDM) [8]; however, a number of shortcomings of OFDM in the MC systems have been noted as well, such as large side lobes of the frequency response of the filters that characterize the subcarrier channels, high peak-to-average power ratio (PAPR), and sensitivity to carrier frequency offset. Compared with OFDM, the ABPSK allows a high level of subchannel spectral containment as Figure 1 shows. So we propose that the ABSPK technique can be used in MC schemes.

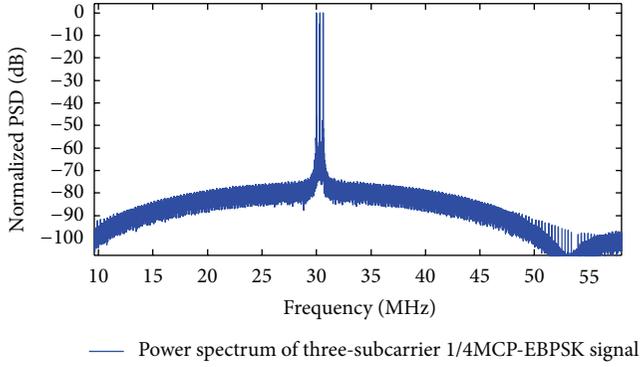


FIGURE 1: Power spectrum of 1/4MCP-EBPSK signal with three subcarriers.

## 2. Uniform Formula of ABPSK

Denote the following.

- (1) Data bit duration  $T$ , it lasts  $N \geq 1$  carrier wave period, that is,  $T = N/f_c$ .
- (2) Phase modulation duration  $\tau$ , it lasts  $K$  carrier wave period, that is,  $\tau = K/f_c$  and  $K \leq N$ .
- (3)  $g_0$  and  $g_1$  are modulation waveforms corresponding to bit “0” and bit “1”, respectively.

Then the uniform formula of ABPSK is defined as

$$g_0(t) = A \sin 2\pi f_c t, \quad 0 \leq t < T,$$

$$g_1(t) = \begin{cases} B \sin(2\pi f_c t \pm \sigma), & 0 \leq t < \tau, \quad 0 \leq \Delta \leq 1, \quad 0 \leq \eta \leq 1, \\ A \sin 2\pi f_c t, & \tau \leq t < T. \end{cases} \quad (1)$$

When the phase changes abruptly,  $\sigma \in [0, \pi]$ ; when phase changes continuously,  $\sigma = \pm \Delta \sin(\eta * 2\pi f_c t)$ . The modulated waveform has the following features.

- (1) Data bit “0” lasts  $N$  carrier cycles in  $T$ , the carrier cycle is  $1/f_c$  and the amplitude is  $A$ .
- (2) Data bit “1” consists of  $K$  carrier cycles with an initial phase  $\sigma$ , amplitude  $B$  and  $N-K$  carrier cycles with the same phase as bit “0”, and amplitude  $A$ .
- (3) If there are no phase changes or amplitude changes, the waveforms are normal sine wave. When  $\sigma \in [0, \pi]$ , it is the EBPSK; when  $\sigma = \pm \Delta \sin(\eta * 2\pi f_c t)$ , according to the sign of  $\Delta$  and value of  $\eta$ , it is the CP-EBPSK and MCP-EBPSK, respectively.
- (4) When  $B = 0$ ,  $\sigma \in [0, \pi]$ , (1) is MCM as has been mentioned in [9]; when  $B = A$ , (1) denote the following.
  - (a) When  $\sigma = \pi$ ,  $\tau = T$ , it represents the classical BPSK; when  $\tau = T/4$ , it represents the 3PRK in [9].
  - (b) When  $\sigma = \pi/2$ ,  $\tau = T/4$ , it represents 3PSK in [9].

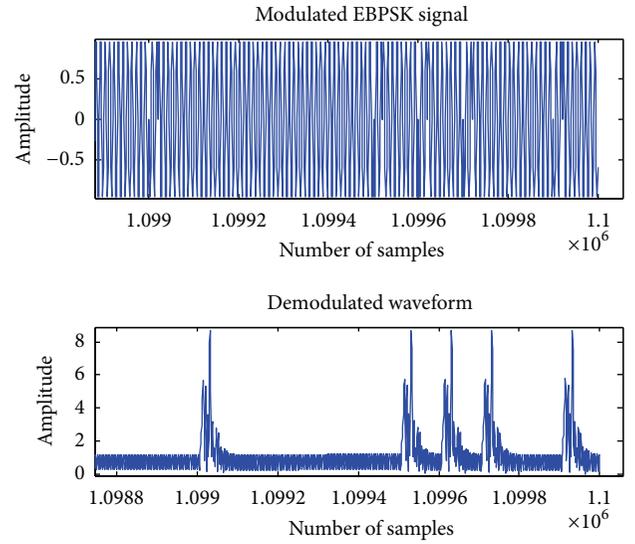


FIGURE 2: Modulated and demodulated waveform of EBPSK.

- (5) The modulation degrees differ according to the phase change  $S$  and modulate duration. The phase changes could control the bandwidth: when the phase vanishingly changes, the spectrum is the narrowest, and when the changes are  $\pi$ , the spectrum is the widest; Denote  $\tau/T = K/N$  as “modulated duty cycle.”

According to (1), choosing a proper value of  $\Delta$ ,  $\eta$ ,  $\sigma$ ,  $\tau$  and using a better shaping-filter could tighten the bandwidth of the modulated signal. We call the signal defined by (1) the uniformly asymmetric binary phase shift keying (ABPSK).

## 3. Special Impacting Filter (SIF)

The demodulation of AEBPSK signal based on the special digital impacting filter is in [10, 11]. The SIF is a kind of digital IIR filters; when working on the proper frequency range, it can convert the phase changes to amplitude impacting. The places where phase is unchanged have much lower impacting than the changed places and most noise is removed at the same time (see Figure 2). Based on the output response, direct amplitude detection can be used for the symbol judgment in intermediate frequency instead of down-conversion to base-band. The filtering mechanism in that filter offers different band to signal and noise, which is discussed in [10, 11] in detail.

Different from normal multicarrier transmission system, the ABPSK subcarriers need not guard interval or satisfy the orthogonal condition. Every subchannel set differs from the others. A narrow pass band of the SIF is used to separate the subcarriers. In order to eliminate the adjacent channel interference, we need to set the notch frequency of the cascade notch filter the same as other carriers frequency to suppress other subchannels’ interference and the output demodulation results [10]. The tiny phase change of the ABPSK makes the demodulation performance closely related to the sampling rate. Only by setting the proper parameter

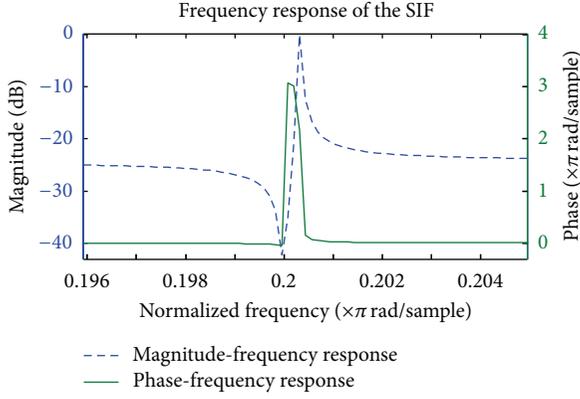


FIGURE 3: Frequency response of the SIF.

of the filter could it reflect the signal change. Based on the above condition, the parameters of the SIF bank correspond with each subcarrier's sampling rate, respectively.

#### 4. Design of Cascade Notch SIF (CNSIF)

The key to demodulate the multicarrier ABPSK signal is the design of the receive filter bank, that is, cascade notch filter based on the SIF. Each subchannel regards the other subchannels as a sinusoidal interference, the notch filter cancelling the interference and SIF demodulating the corresponding signal. Many methods have been mentioned in [10, 11] for the SIF design and performance analysis, but the SIF pole radius is specified by the designer arbitrarily. Here we present a new approach to the numerical solution of the SIF as well as use an iterative scheme to gain the pole angle of the notch filter that fulfills the cost function.

**4.1. Calculation of the SIF Pole Radii.** Consider an SIF with one pair of conjugate zero-poles whose transfer function is

$$H(z) = \frac{(1 - r_z e^{j\omega_c - \epsilon} z^{-1})(1 - r_z e^{-j\omega_c - \epsilon} z^{-1})}{(1 - r_p e^{j\omega_c + \mu} z^{-1})(1 - r_p e^{j\omega_c + \mu} z^{-1})}. \quad (2)$$

Different from the conventional IIR filters, the zero angles are less than the corresponding carrier frequency and are less than the pole angles. Place the zeros on the unit circle with angle shift  $\nu$ , that is,  $r_z = 1$ ,  $\omega_z = \omega_c - \nu$ ; the poles are close to the unit circle and are near to zeros with angle shift  $\mu$ , that is,  $0 \ll r_p < 1$ ,  $\omega_p = \omega_c + \mu$ . Generally the carrier frequency is situated in the maximum slope of the magnitude response. Figure 3 shows the amplitude-frequency response and phase-frequency response of the demodulation impacting filter with one pair of conjugate zero-poles, the normalized carrier frequency is  $0.2\pi$  rad/sample; it has narrow notch-frequency-selecting performance near the carrier frequency.

According to (2), we can evaluate the frequency response with graphical method (see Figure 4). In practice the poles and zeros are even much closer than Figure 4; the poles are

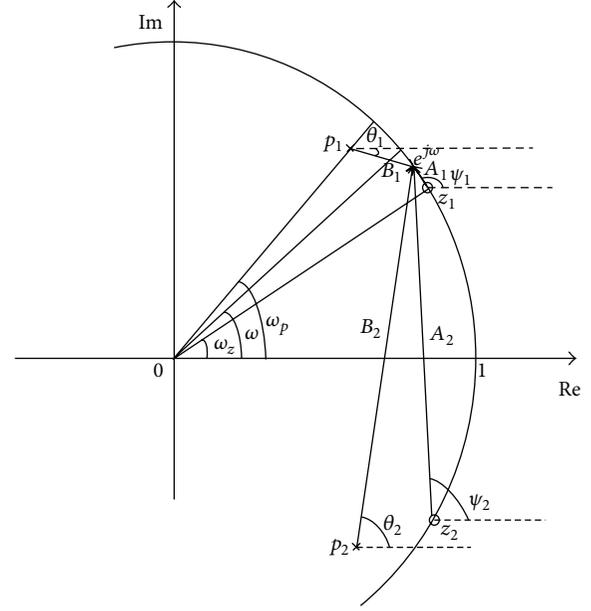


FIGURE 4: Schematic of zero-pole relative position.

even much closer to the unit circle. In the  $z$ -plane, the zero-pole vectors shift on the unit circle  $e^{j\omega}$ .

$z_1, z_2$  are zeros lying on the unit circle;  $p_1, p_2$  are poles near the unit circle and zeros.  $A_1, \psi_1, A_2, \psi_2, B_1, \theta_1, B_2, \theta_2$  represent the module and angle of  $z_1, z_2, p_1, p_2$ , respectively. For the periodic frequency characteristic of the digital filters, we could just discuss the situation of  $0 < \omega < \omega_s/2$ , that is,  $0 \sim \pi$ .

From Figure 4, the frequency magnitude response of (2) can be rewritten as

$$|H(e^{j\omega})| = \frac{A_1 \cdot A_2}{B_1 \cdot B_2}. \quad (3)$$

Because the zeros and poles are close to the unit circle and the distances from the conjugate pole and zero to  $e^{j\omega}$  are nearly equal when they shift in the region of  $0 < \omega < \omega_z$  and  $\omega_p < \omega < \omega_s/2$ ; that is,  $A_1 \approx B_1, A_2 \approx B_2$ . So

$$|H(e^{j\omega})| = 1, \quad 0 < \omega < \omega_z \cup \omega_p < \omega < \frac{\omega_s}{2}. \quad (4)$$

But when they shift in the region of  $\omega_z < \omega < \omega_p$ , the distance  $A_1, B_1$  is much smaller than the distance  $A_2, B_2$ ; we may state that  $A_2 \approx B_2$ . The frequency magnitude response can be written as

$$|H(e^{j\omega})| = \frac{A_1}{B_1}, \quad \omega_z < \omega < \omega_p. \quad (5)$$

Enlarge the region of  $\omega_z < \omega < \omega_p$ , the relative position between zero-pole vector and  $e^{j\omega}$  depicted in Figure 5, here the influence of the complex conjugate zero-pole pair will be negligible.

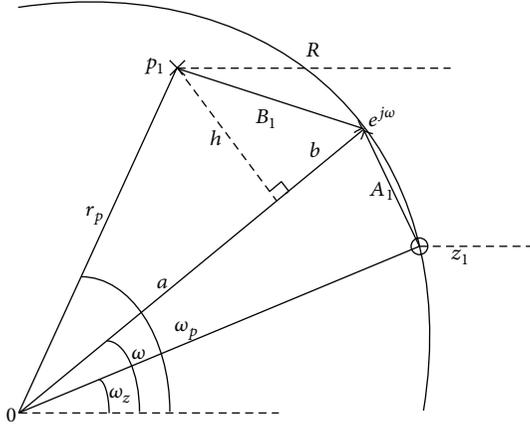


FIGURE 5: Schematic of the enlarged relative position between zero-pole.

With the Pythagorean theorem, we obtain two equations:

$$A_1 = 2 \sin\left(\frac{\omega - \omega_z}{2}\right), \quad (6)$$

$$B_1 = \sqrt{h^2 + b^2} = \sqrt{r_p^2 - a^2 + b^2}, \quad (7)$$

where  $r_p$  is the module of pole  $p_1$ .

Substituting  $a = \sqrt{r_p^2 - r_p^2 \sin^2(\omega_p - \omega)}$ ,  $a + b = 1$ , into (7) leads to

$$B_1 = \sqrt{r_p^2 - 2r_p \cos(\omega_p - \omega) + 1}. \quad (8)$$

So the frequency magnitude response in the region of  $\omega_z < \omega < \omega_p$  can be written as

$$|H(e^{j\omega})| = \frac{2 \sin[(\omega - \omega_z)/2]}{\sqrt{r_p^2 - 2r_p \cos(\omega_p - \omega) + 1}}, \quad \omega_z < \omega < \omega_p. \quad (9)$$

According to Mooror's bandwidth definition [12], if the resonance gain over the DC gain exceeds 6 dB in absolute value, the frequencies at 3 dB below/above the peak/notch is the band edges, that is

$$|H(e^{j(\omega_c - B_w/2)})| = \frac{1}{\sqrt{2}} \cdot |H(e^{j\omega_c})|, \quad (10)$$

where  $B_w$  is the bandwidth of the SIF; (10) can be expressed as

$$\frac{\sin[(\omega_p - \omega_z)/2]}{\sin[(\omega_p - B_w/2 - \omega_z)/2]} \cdot \frac{\sqrt{r_p^2 - 2r_p \cos(B_w/2) + 1}}{1 - r_p} = \sqrt{2}. \quad (11)$$

To ensure the impacting effect, the pole and zero angle is specified by the designer. When the bandwidth  $B_w$  is known,

the results of the following two equations are constants, respectively, that is

$$\begin{aligned} \omega_p - \omega_z &= \phi, \\ \frac{2[1 - \cos(\phi - B_w/2)]}{1 - \cos\phi} &= M. \end{aligned} \quad (12)$$

Substituting (12) into (11), we can readily obtain

$$\begin{aligned} r_p &= \\ &= \frac{M - \cos(B_w/2) \pm \sqrt{\cos^2(B_w/2) - 2M \cos(B_w/2) + 2M - 1}}{M - 1}. \end{aligned} \quad (13)$$

The poles and zeros are very close to each other, as a consequence,  $\phi$  and  $B_w$  will be relatively small. To avoid  $r_p > 1$ , we abandon the adding situation. After some further simplification. The final result is

$$\begin{aligned} r_p &= \\ &= \frac{M - \cos(B_w/2) - \sqrt{\cos^2(B_w/2) - 2M \cos(B_w/2) + 2M - 1}}{M - 1}. \end{aligned} \quad (14)$$

Equation (14) shows that the pole radii of SIF can be given in a quantitative way instead of arbitrary setting.

The procedure of the one-pair conjugate zero-poles SIF design could ensure the impacting effect, and we know that if the zeros and poles are far away from the unit circle, they will only affect the amplitude of the frequency response, but not the shape. From this standpoint, we can add more conjugate poles to tighten the bandwidth of the SIF and gain higher impacting effect. Generally we can add two conjugate poles whose angles are equal to the initial poles and are far away from the unit circle.

**4.2. The Cascade Notch SIF (CNSIF).** The multicarrier ABPSK scheme uses the CNSIF banks to separate and demodulate each subcarrier [13]. The key of the CNSIF is the narrow notch-frequency-selecting performance. When the overlapping signals pass the filter banks, the primary signal is enhanced and the noises or interferences are attenuated. Here the SIF enhances the primary signal and the notch filter decreases the disturbing signal. Let us start with the design of the notch filter in details.

Without loss of generality, we assume that the notch filter has  $k$  notch frequencies; the transfer function is given by

$$H_1(z) = \frac{B(z)}{A(z)} = \frac{\prod_{i=1}^k (1 - 2 \cos(\omega_{Ni}) z^{-1} + z^{-2})}{\prod_{i=1}^k (1 - 2r_i \cos(\omega_{pi}) z^{-1} + r_i^2 z^{-2})}, \quad (15)$$

where  $r_i$  is the pole radius. For stability and according to system requirements, the values of  $\omega_{Ni}$  and  $r_i$  are given by the designer ( $0 \leq r_i < 1$ ). Meanwhile the zeros are constrained to

locate on the unit circle at the notch frequencies  $\omega_{Ni}$ , so the problem of designing a notch filter reduces to find the optimal value of  $\omega_{pi}$ .

Let  $a_i = 2 \cos(\omega_{pi})$ , then  $\omega_{pi} = \arccos(a_i/2)$ ,  $-2 \leq a_i \leq 2$ ,  $-\pi \leq \omega_{pi} \leq \pi$ . The transfer function can be rewritten as

$$H_1(z) = \frac{\prod_{i=1}^k (1 - b_i z^{-1} + z^{-2})}{\prod_{i=1}^k (1 - r_i a_i z^{-1} + r_i^2 z^{-2})}, \quad (16)$$

where  $b_i = 2 \cos(\omega_{Ni})$ ,  $i = 1, 2, \dots, k$ .

For convenience, let  $i = 1$ , then the notch frequency is  $\omega_{N1}$ , pole radius is  $r_1$ , and (16) can be expressed as

$$H_1(z) = \frac{B(z)}{A(z)} = \frac{1 - b_1 \cdot z^{-1} + z^{-2}}{1 - a_1 r_1 z^{-1} + r_1^2 z^{-2}}. \quad (17)$$

Assuming that  $\omega_z < \omega_p < \omega_{N1} < \omega_{N2} < \dots < \omega_{Ni}$ ,  $\omega_{N1} \geq \omega_p + B_w/2$ . With a given weighting function  $W(\omega)$ , we can define the cost function as

$$C(a) = \int_R W(\omega) |H(e^{j\omega}) - H_1(e^{j\omega})|^2 d\omega, \quad (18)$$

where the region  $R = [\omega_p, \omega_{N1} - \delta] \cup [\omega_{N1} + \delta, \pi]$ ,  $\delta$  is a prescribed small positive number, combining (4) and (18) yields

$$C(a) = \int_R W(\omega) |1 - H_1(e^{j\omega})|^2 d\omega. \quad (19)$$

To ensure the impacting characteristic of SIF and cancel the signal disturbing, the cascade notch filter should not affect the frequency response of the SIF; that is, the cost function should be minimized.

Substituting (17) into (19), we can obtain

$$\begin{aligned} C(a_1) &= \int_R W(\omega) \left| 1 - \frac{B(e^{j\omega})}{A(e^{j\omega})} \right|^2 d\omega \\ &= \int_R \frac{W(\omega)}{|A(e^{j\omega})|^2} |A(e^{j\omega}) - B(e^{j\omega})|^2 d\omega \\ &= \int_R \frac{W(\omega)}{|A(e^{j\omega})|^2} \left| [(r_1^2 - 1)e^{-2j\omega} + b_1 e^{-j\omega}] - r_1 a_1 e^{-j\omega} \right|^2 d\omega. \end{aligned} \quad (20)$$

Let  $W(\omega)/|A(e^{j\omega})|^2$  be the new weighting function, since  $r_1$  and  $b_1$  are known numbers, we need to find  $a_1$  to minimize the cost function. For  $a_1 = 2 \cos(\omega_{p1})$ ,  $\omega_{N1} - \delta \leq \omega_{p1} \leq \omega_{N1} + \delta$ , we can use an iteration scheme to find the optimal value of  $a_1$ ; each iteration refreshes the value of  $A(e^{j\omega})$ . We

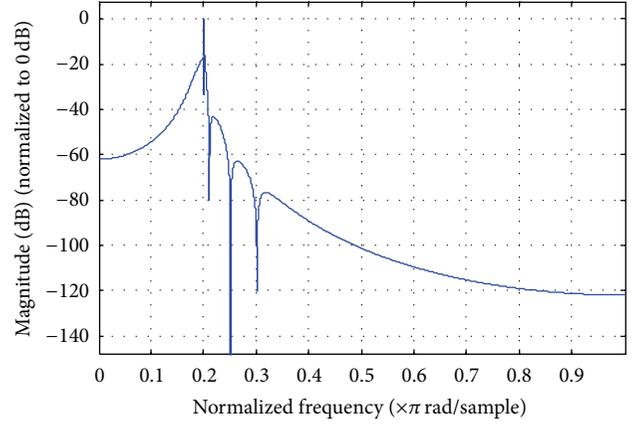


FIGURE 6: The frequency magnitude response of the designed CNSIF.

denote the value of  $a_1$  at the  $n$ th iteration as  $a_{1n}$ ,  $A(e^{j\omega})$  as  $A_n(e^{j\omega})$ , similarly. When  $A(e^{j\omega})$  gets the new value, the weighting function is updated immediately.

For convenience, substitute

$$p(\omega) \leftarrow (r_1^2 - 1)e^{-2j\omega} + b_1 e^{-j\omega}, \quad q(\omega) \leftarrow -r_1 e^{-j\omega}$$

$$\begin{aligned} \Rightarrow C(a_{1n}) &= \int_R \frac{W(\omega)}{|A_n(e^{j\omega})|^2} |p(\omega) + q(\omega) a_{1n}|^2 d\omega \\ &= \int_R \frac{W(\omega)}{|A_n(e^{j\omega})|^2} \\ &\quad \cdot [a_{1n}^2 |q(\omega)|^2 + 2a_{1n} |p(\omega) q(\omega)| \\ &\quad + |p(\omega)|^2] d\omega \\ &= \int_R \frac{W(\omega)}{|A_n(e^{j\omega})|^2} [a_{1n}^2 |q(\omega)|^2 \\ &\quad + 2a_{1n} \cdot \text{Re}(p(\omega) q^*(\omega)) \\ &\quad + |p(\omega)|^2] d\omega. \end{aligned} \quad (21)$$

After some simplification, (21) is rewritten as

$$\begin{aligned} C(a_{1n}) &= \int_R W_n(\omega) [|q(\omega)|^2 \cdot a_{1n}^2 + 2 \cdot \text{Re}(p(\omega) q^*(\omega)) \\ &\quad \cdot a_{1n} + |p(\omega)|^2] d\omega, \end{aligned} \quad (22)$$

where

$$W_n(\omega) = \frac{W(\omega)}{|A_{n-1}(e^{j\omega})|^2}. \quad (23)$$

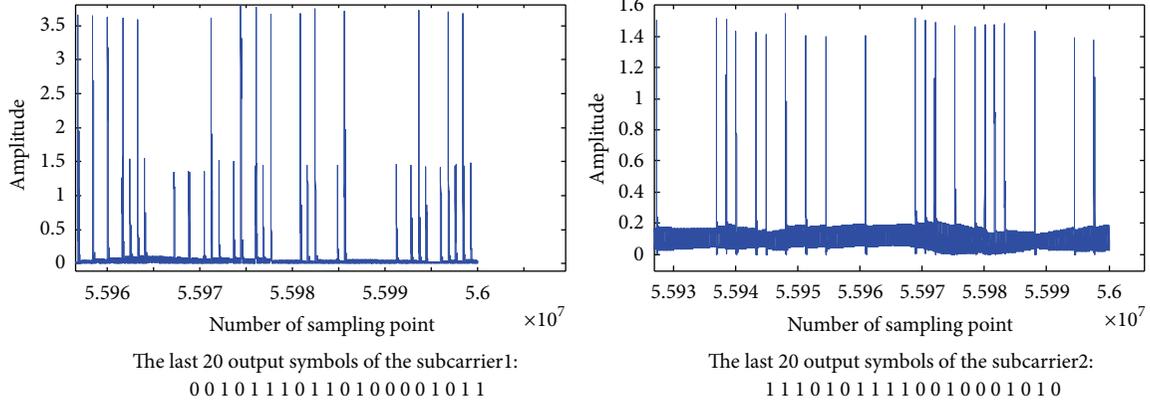


FIGURE 7: The waveform after the CNSIF bank filtering.

So we can determine the value of  $a_{1n}$  by solving the constrained optimization problem

$$\begin{aligned} \min \quad & \int_R W_n(\omega) d\omega \cdot \left[ |q(\omega)|^2 \cdot a_{1n}^2 + 2 \cdot \text{Re}(p(\omega) q^*(\omega)) \right. \\ & \left. \cdot a_{1n} + |p(\omega)|^2 \right] \\ \text{s.t.} \quad & 0 \leq r_1 < 1 \\ & -2 \leq a_{1n} \leq 2 \\ & \omega_{N1} - \delta \leq \omega_{p1} \leq \omega_{N1} + \delta. \end{aligned} \quad (24)$$

We can see that the cost function is a parabola with its mouth opened up, so it could be transformed to a closed form and have an optimal solution. At the  $n$ th iteration, minimizing (22) subjected to constraints of (24) is a typical quadratic programming problem. We can compute the solution efficiently.

## 5. Design Example

**5.1. Design Parameters.** Set the normalized three notch frequencies as  $0.21\pi$ ,  $0.25\pi$ , and  $0.3\pi$ . Easily derive the initial values that  $a_1 = 0.521292$ ,  $a_2 = 0.555360$ ,  $a_3 = 0.587785$ . Let  $\delta = 0.001\pi$ ; the convergent values are  $0.52206411$ ,  $0.55613211$ , and  $0.58855711$ . Given the following specifications:  $r_1 = r_2 = r_3 = 0.95$ ; weighting function  $W(\omega) = 1$ ;  $\omega_z = 0.6282$  rad,  $\omega_p = 0.62855$  rad,  $B_w = \omega_p - \omega_z = 0.00035$  rad ( $1.114e^{-4}\pi$  rad). The pole radii  $r_p$  defined in (14) is  $0.999999969375$ . Adding another three pairs of conjugate poles with angle  $0.62855$  rad, radii are  $0.9$ ,  $0.92$ , and  $0.95$ , respectively. The frequency magnitude response of the final designed CNSIF is shown in Figure 6.

**5.2. Performance.** We choose the MCP-EBPSK signal as a source, simulating the performance of two subcarriers. The parameters are as follows:  $f_{c1} = 10$  MHz,  $f_{c2} = 10.04$  MHz,  $f_{s1} = 160$  MHz,  $f_{s2} = 100.4$  MHz,  $A = 1$ ;  $B = 1$ ;  $\Delta = 0.1$ ;  $\eta = 0.5$ ;  $K = 2$ ;  $N_1 = 100$ ;  $N_2 = 160$ ; the entire test symbols are

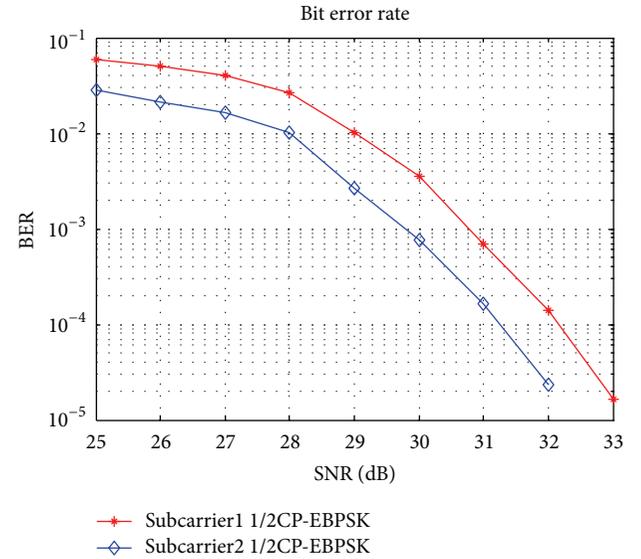


FIGURE 8: BER performances of two subcarriers.

3 million. Set the notch frequencies as  $0.1255\pi$  and  $0.1992\pi$ , we can obtain  $a_1 = 1.84656$ ,  $a_2 = 1.62098$ , after six iterations converge to  $1.84855455$  and  $1.6219734$ , respectively. Let  $r_1 = r_2 = 0.95$ ,  $\delta = 0.001\pi$ , weighting function  $W(\omega) = 1$ .

We can obtain the parameters of the CNSIF bank as

$$\begin{aligned} a_1 &= [1, -6.09611187753762, 16.418167515832877, \\ & \quad -25.326641789930786, 24.298560015038895, \\ & \quad -14.721898165092053, 5.447828484314318, \\ & \quad -1.116449367584231, 0.098333159057181]; \\ b_1 &= [0, 0, 0, 0, 1, -3.694313616799019, 5.411987912102056, \\ & \quad -3.694313616799019, 1]; \\ a_2 &= [1, -5.406525327518239, 13.652579283244823, \\ & \quad -20.517799618098030, 19.861352592576445, \\ & \quad -12.536073668382514, 4.996805854728122, \\ & \quad -1.139889481405723, 0.120047843082279]; \\ b_2 &= [0, 0, 0, 0, 1, -3.239006288658281, 4.622788277906668, \\ & \quad -3.239006288658281, 1]. \end{aligned}$$

Figure 7 shows the waveform of the component signal after the CNSIF bank filtering. We can see that the designed CNSIF could separate the component MCP-EBPSK signals. The BER performance is shown in Figure 8.

## 6. Conclusions

With the intuitive method known as zero-pole placement, we analyzed the frequency response of the SIF in a quantitative way. The equation about the pole radii under specified bandwidth is derived. The designed CNSIF could not only maintain the impacting characteristic but also attenuate the interchannel interference (ICI). From Figure 6, we can also see that adding notch zeros could speed up the decaying of the response curve and tighten the bandwidth; this will help to filter the noise. Compared with OFDM, the multicarrier scheme of ABPSK could have quite low PAPR ratio. It can also be combined with MIMO, cognitive radio, and so forth to get better performance, which is our next work to emphasize.

## Acknowledgments

The authors thank all of the reviewers for their valuable comments, which have considerably helped in improving the overall quality of the work presented in the revised paper. Editorial assistance and proofreading of the manuscript from Bingcheng Zhu are appreciated. This work is supported by the State 863 Project (2008AA01Z227), the National Natural Science Foundation of China (NSFC), under the Grant 61271204.

## References

- [1] H. R. Walker, "High Speed Binary Data Communication System [P]," US Patent 4742532, 1988.
- [2] H. R. Walker, "VPSK and VMSK modulation transmit digital audio and video at 15 Bits/Sec/Hz," *IEEE Transactions on Broadcasting*, vol. 43, no. 1, pp. 96–103, 1997.
- [3] H. R. Walker and B. Stryzak, "Comparing ultra wideband and ultra narrow band modulation," in *Proceedings of the International Conference on Computing, Communications and Control Technologies (CCCT '04)*, Austin, Tex, USA, 2004.
- [4] L. Wu, "The report of recent developments in UNB high-speed communications," *Progress in Nature Science*, vol. 17, no. 11, pp. 1467–1473, 2007.
- [5] M. Feng, *Research on key problems in high-efficiency modulation [Ph.D. thesis]*, Southeast University, Nanjing, China, 2008.
- [6] F. He and L. Wu, "Analysis of power spectrum of continuous phase waveforms for binary modulation communications," in *Proceedings of the International Conference on Ultra Modern Telecommunications and Workshops*, Petersburg, Russia, October 2009.
- [7] Y. Jin, L. Wu, A. Feng, and D. Lei, "An M-ary and continuous phase modem with random-polar," *Chinese Journal of Radio Science*, vol. 27, no. 5, pp. 931–936, 2012 (Chinese).
- [8] T. A. Weiss and F. K. Jondral, "Spectrum pooling: an innovative strategy for the enhancement of spectrum efficiency," *IEEE Communications Magazine*, vol. 42, no. 3, pp. S8–S14, 2004.
- [9] H. R. Walker, *Ultra Narrow Band Modulation Textbook*, 2010, <http://www.vmsk.org/>.
- [10] M. Feng, P. Gao, and L. Wu, "Analysis and simulation of special filtering based on ultra narrow band modulated signal," *Journal of Southeast University*, vol. 40, no. 2, pp. 227–230, 2010 (Chinese).
- [11] P. Gao, *On impacting filter in unb receiver [M.S. thesis]*, Southeast University, Nanjing, China, 2010.
- [12] J. A. Moorer, "The manifold joys of conformal mapping: applications to digital filtering in the studio," *Journal of the Audio Engineering Society*, vol. 31, no. 11, pp. 826–841, 1983.
- [13] Z. Chen, L. Wu, and X. Chen, "The impacting filter bank design for MCP-EBPSK multicarrier demodulation," *Signal Processing*, vol. 28, no. 8, pp. 1063–1068, 2012 (Chinese).

## Research Article

# Fault Diagnosis for Wireless Sensor by Twin Support Vector Machine

Mingli Ding, Dongmei Yang, and Xiaobing Li

*Department of Automatic Test and Control, Harbin Institute of Technology, Harbin, Heilongjiang 150001, China*

Correspondence should be addressed to Mingli Ding; [mmingli@126.com](mailto:mmingli@126.com)

Received 17 January 2013; Revised 20 March 2013; Accepted 7 April 2013

Academic Editor: Saeed Balochian

Copyright © 2013 Mingli Ding et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Various data mining techniques have been applied to fault diagnosis for wireless sensor because of the advantage of discovering useful knowledge from large data sets. In order to improve the diagnosis accuracy of wireless sensor, a novel fault diagnosis for wireless sensor technology by twin support vector machine (TSVM) is proposed in the paper. Twin SVM is a binary classifier that performs classification by using two nonparallel hyperplanes instead of the single hyperplane used in the classical SVM. However, the parameter setting in the TSVM training procedure significantly influences the classification accuracy. Thus, this study introduces PSO as an optimization technique to simultaneously optimize the TSVM training parameter. The experimental results indicate that the diagnosis results for wireless sensor of twin support vector machine are better than those of SVM, ANN.

## 1. Introduction

In the past years, various data mining techniques including artificial neural networks have been applied to fault diagnosis for wireless sensor because they have the advantages of discovering useful knowledge from large data sets [1–5]. Though fault diagnosis for wireless sensor based on artificial neural networks can show encouraging results, there are also many problems that need to be solved, such as local optimization and overfitting in the artificial neural networks [6–11]. Support vector machine (SVM), based on structure risk minimization principle can use nonlinear mapping to transform an input space to a high-dimension space based on an internal integral function and then looks for a nonlinear relationship between inputs and outputs in that space [12–14]. SVM can find global optimum solutions for problems with small training samples, high dimensions, nonlinear [15, 16]. Twin SVM is a binary classifier that performs classification by using two nonparallel hyperplanes instead of the single hyperplane used in the classical SVM. However, the choice of the training parameters has a heavy impact on the classification accuracy of twin support vector machine. Particle swarm optimization is an evolutionary computation technique, which is inspired by social behavior among

individuals. Thus, particle swarm optimization is used to optimize the TSVM parameters.

In the study, a novel classification method by twin support vector machine (PSO-TSVM) is proposed to fault diagnosis for wireless sensor, where particle swarm optimization is to find the optimal settings of parameters of SVM. Then, we collect 260 samples to study the diagnosis performance of twin support vector machine classifier, where 170 of them are used to train the diagnosis model of twin support vector machine classifier, and others are used to test the diagnosis performance of twin support vector machine classifier. The experimental results indicate that the diagnosis results for wireless sensor of twin support vector machine are better than those of SVM, ANN.

## 2. Twin Support Vector Machine

Based the Karush-Kuhn-Tucker theorem of optimization theory [17, 18], the nonlinear decision function is:

$$f(x) = \text{sign} \left( \sum_{i=1}^N \alpha_i y_i K(x_i, x_j) + b \right). \quad (1)$$

The most commonly used kernel function is the radial basis function (RBF) kernel, which can be reproduced as follows:

$$K(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|}{2\sigma^2}\right), \quad (2)$$

where  $\sigma$  is a positive real number.

The nonlinear TWSVM seeks two nonparallel hyperplane in  $R^n$ :

$$\begin{aligned} w_1' \cdot \phi(x) + b_1 &= 0, \\ w_2' \cdot \phi(x) + b_2 &= 0. \end{aligned} \quad (3)$$

For finding the hyperplanes, it is required to get the solutions to the primal problems.

Minimize

$$U(w_1, \xi_2) = \frac{1}{2} \|Aw_1 + e_1 b_1\|^2 + c_1 e_2' \xi_2 \quad (4)$$

subject to

$$\begin{aligned} -(Bw_1 + e_2 b_1) + \xi_2 &\geq e_2, \\ \xi_2 &\geq 0, \\ c_1 &> 0. \end{aligned} \quad (5)$$

And minimize

$$U'(w_2, \xi_1) = \frac{1}{2} \|Bw_2 + e_2 b_2\|^2 + c_2 e_1' \xi_1 \quad (6)$$

subject to

$$\begin{aligned} (Aw_2 + e_1 b_2) + \xi_1 &\geq e_1, \\ \xi_1 &\geq 0, \\ c_2 &> 0, \end{aligned} \quad (7)$$

where  $c_1, c_2$  are the punishment parameters and  $e_1, e_2$  are vectors of ones of appropriate dimensions.

### 3. Parameters Optimization of TSVM by PSO

Particle swarm optimization is an evolutionary computation technique, which is inspired by social behavior among individuals. Each particle moves in the direction of its previously best position and its best global position during each generation [19–21]. Thus, particle swarm optimization is used to optimize the TSVM parameters.

In the study, we use the RBF kernel function for the TSVM classifier because the RBF kernel function can analyze higher-dimensional data, and TSVM with RBF kernel function only has two parameters,  $C$  and  $\sigma$  determined. Therefore, the particle is comprised of two parts,  $C$  and  $\sigma$ , when the RBF kernel is selected. The process of optimizing the TSVM parameters with PSO can be summarized as follows.

*Step 1.* Randomly generate initial population, initial particle and initial velocity.

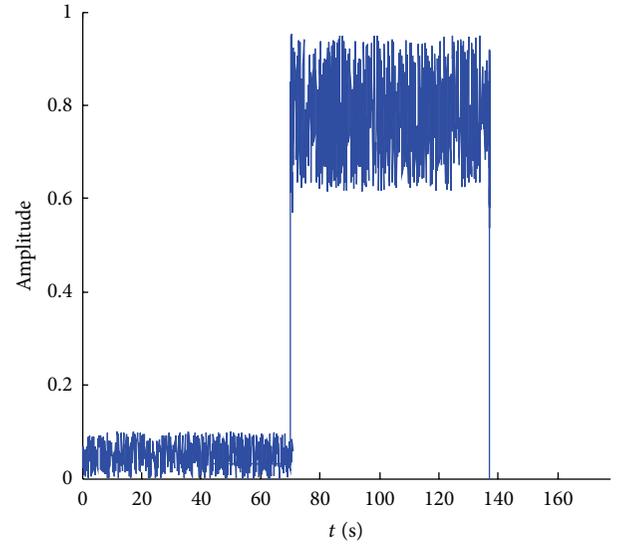


FIGURE 1: The output signal of shock failure.

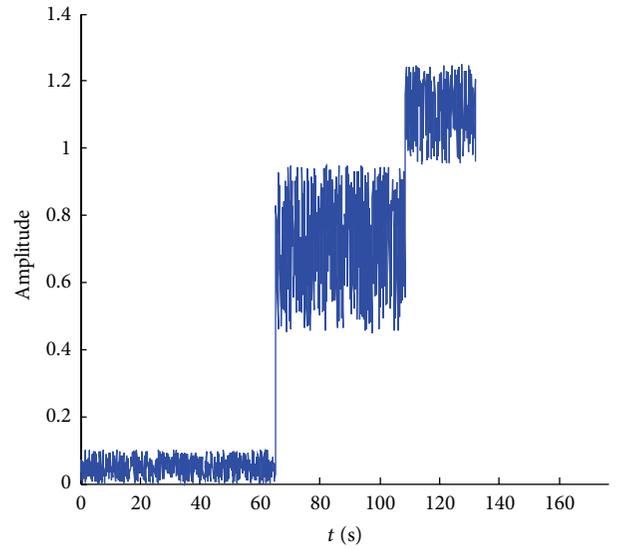


FIGURE 2: The output signal of biasing failure.

*Step 2.* Set the learning parameters  $c_1$  and  $c_2$ , the inertia weight  $\omega$ , and the maximum number of iterations.

*Step 3.* Fitness evaluation: the fitness function is defined as the following formula:

$$\text{Fitness} = \left(1 - \frac{T}{F + T}\right) \times 100\%, \quad (8)$$

where  $T$  denotes the correct classification and  $F$  denotes the false classification.

*Step 4.* Update velocity and position of the particle.

*Step 5.* If maximum iterations predefined are met, the program is stopped. Otherwise, go to Step 3.

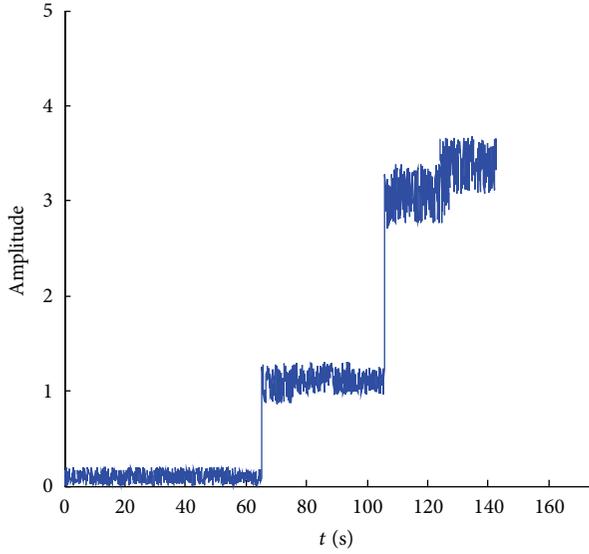


FIGURE 3: The output signal of short-circuit failure.

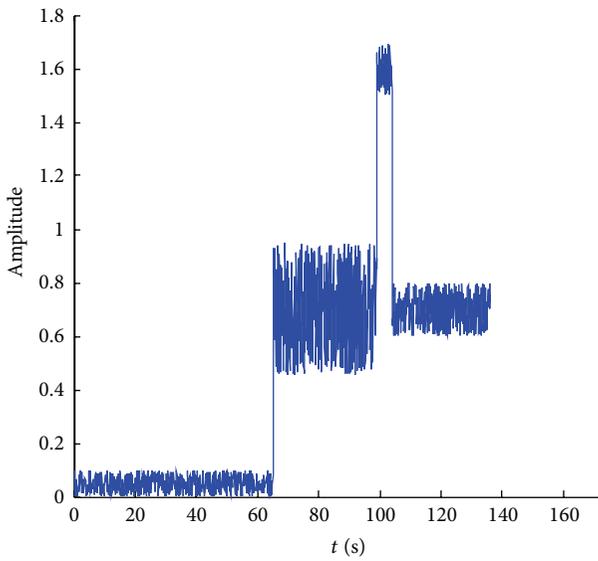


FIGURE 4: The output signal of shifting failure.

#### 4. Experimental Study for Fault Diagnosis of Wireless Sensor

In the study, the four fault types of wireless sensor including shock, biasing, short circuit, and shifting are applied to test the diagnosis ability of TSVM compared with other diagnostic methods. The normal data belongs to class 1, shock belongs to class 2, biasing belongs to class 3, short circuit belongs to class 4, and shifting belongs to class 5. The typical output signals of the above four fault types of wireless sensor can be described in Figures 1, 2, 3, and 4, respectively.

The values of the features and the corresponding state types of wireless sensor are used to train twin support vector machine classifier. In the study, we collect 260 samples to

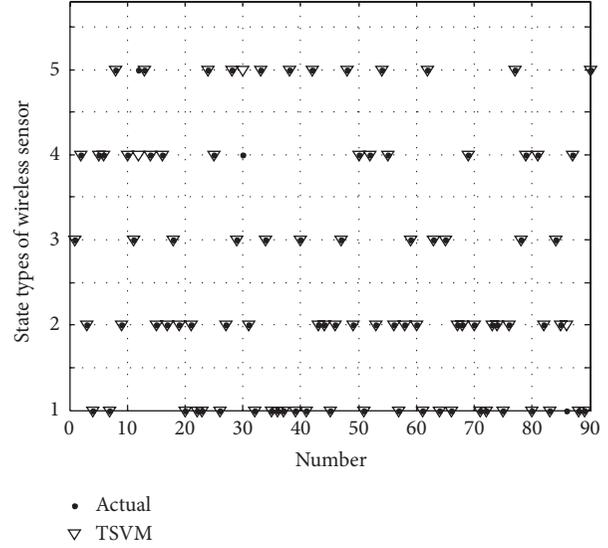


FIGURE 5: The diagnosis results of twin support vector machine.

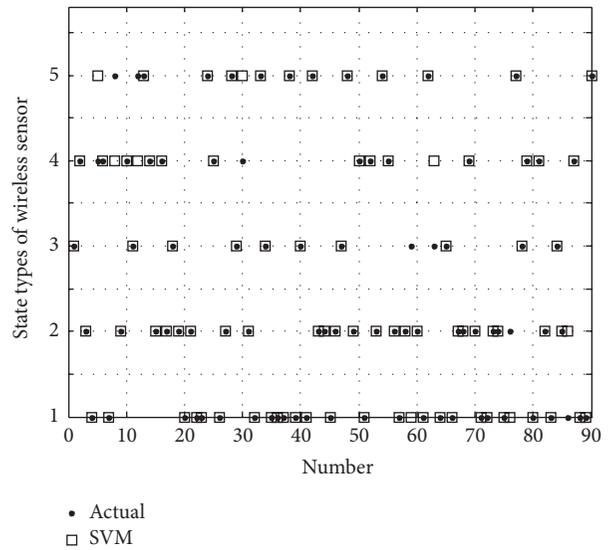


FIGURE 6: The diagnosis results of support vector machine.

study the diagnosis performance of twin support vector machine classifier, where 170 of them are used to train the diagnosis model of twin support vector machine classifier, and others are used to test the diagnosis performance of twin support vector machine classifier. Some of the experimental data are given in Table 1.

Figure 5 gives the diagnosis results of twin support vector machine, state types of wireless sensor including normal state, shock, biasing, short circuit, and shifting are given in Figure 5, which are denoted as 1~5, respectively; Figure 6 gives the diagnosis results of the support vector machine; Figure 7 gives the diagnosis results of artificial neural network. The number of incorrect diagnosis of TSVM, SVM, and ANN is 96.7, 91.1, 83.3, respectively. The comparison of the diagnosis results for wireless sensor among TSVM, SVM,

TABLE 1: The experimental data.

$X_1$	$X_2$	$X_3$	$X_4$	$X_5$	$X_6$	$X_7$	$X_8$	$X_9$	$X_{10}$	Fault type	
0.1461	0.1219	0.7039	1.754	0.7042	0.7212	0.6884	0.7134	0.6879	0.6333	Shock	
0.1457	0.1227	0.7033	1.605	0.7122	0.6991	0.7017	0.7121	0.6671	0.6748		
0.1462	0.1225	0.7052	1.669	0.7001	0.712	0.6657	0.7145	0.67	0.6501		
0.1463	0.1226	0.7044	1.590	0.7183	0.7056	0.6775	0.7043	0.681	0.6630		
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮		
0.1458	0.1221	0.6988	1.609	0.7211	0.7255	0.7011	0.7126	0.7028	0.684		
0.1461	0.1228	0.7052	1.603	0.6847	0.6939	0.6507	0.6802	0.6613	0.6217		
0.1460	0.1221	0.7045	1.713	0.7137	0.6977	0.6670	0.6930	0.6741	0.643		
0.1462	0.1223	0.6997	1.119	1.149	1.164	1.164	1.148	1.18	1.161		Biasing
0.1460	0.1222	0.7027	1.137	1.153	1.107	1.149	1.159	1.166	1.179		
0.1458	0.1221	0.6988	1.126	1.137	1.1	1.172	1.146	1.172	1.181		
0.1459	0.1224	0.7051	1.153	1.164	1.124	1.171	1.164	1.161	1.163		
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮		
0.1461	0.1228	0.7052	1.201	1.357	1.035	1.192	1.169	1.161	1.159		
0.1460	0.1221	0.7045	1.163	1.167	1.122	1.153	1.144	1.172	1.169		
0.1459	0.1224	0.7051	0.1853	0.0038	0.0053	0.001	0.00082	0.0126	0.0086	Short circuit	
0.1462	0.1225	0.7052	0.1752	0.0016	0.0031	0.0012	0.0012	0.0115	0.0064		
0.1460	0.1221	0.7045	0.1755	0.0003	0.0055	0.0017	0.0009	0.0127	0.0063		
0.1458	0.1221	0.6988	0.1849	0.0011	0.0049	0.0013	0.00115	0.0086	0.0078		
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮		
0.1462	0.1225	0.7052	0.1752	0.0016	0.0031	0.0012	0.0012	0.00115	0.0064		
0.1461	0.1228	0.7052	0.1855	0.0002	0.0012	0.001	0.0088	0.0096	0.0082		
0.1460	0.1221	0.7045	3.546	3.786	4.046	4.183	4.492	4.673	4.942		Shifting
0.1462	0.1225	0.7052	3.581	3.812	4.069	4.156	4.438	4.619	4.923		
0.1459	0.1224	0.7051	3.527	3.846	4.057	4.191	4.493	4.615	4.879		
0.1460	0.1222	0.7027	3.588	3.819	4.051	4.168	4.513	4.692	4.851		
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮		
0.1457	0.1227	0.7033	3.593	3.779	4.053	4.163	4.472	4.701	4.928		
0.1458	0.1221	0.6988	3.649	3.798	4.055	4.161	4.485	4.688	4.899		

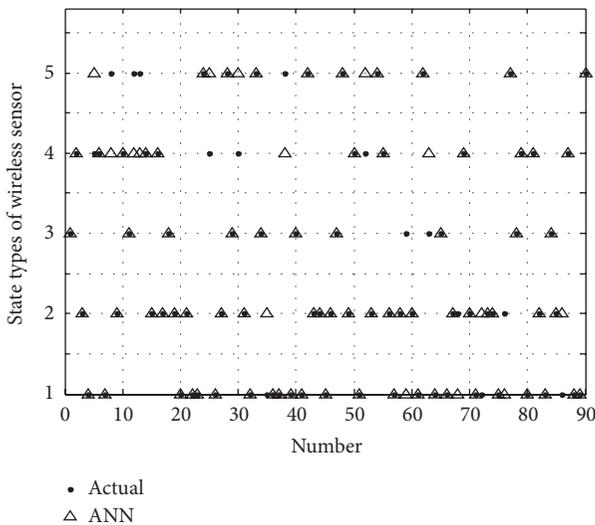


FIGURE 7: The diagnosis results of artificial neural network.

TABLE 2: The comparison of the diagnosis results for wireless sensor among the three classifiers.

Classifier	The number of incorrect diagnosis	Diagnosis accuracy/%
TSVM	3	96.7
SVM	8	91.1
ANN	15	83.3

and ANN is given in Table 2. Then, we can conclude that the diagnosis results of twin support vector machine are better than those of SVM and ANN in the fault diagnosis of wireless sensor.

### 5. Conclusion

A novel classification method by twin support vector machine (TSVM) is proposed to fault diagnosis for wireless

sensor in this paper, where PSO is to find the optimal settings of parameters in SVM. In the study, the four fault types of wireless sensor including shock, biasing, short circuit, and shifting are applied to test the diagnosis ability of TSVM compared with other diagnostic methods. The experimental results indicate that the diagnosis results for wireless sensor of twin support vector machine are better than those of SVM and ANN.

## Acknowledgments

This work is supported by the National Natural Science Foundation of China (Grants no. 60901042 and no. 61171196).

## References

- [1] Z. You, X. Zhao, H. Wan, N. N. H. William, Y. Wang, and M. Gu, "A novel fault diagnosis mechanism for wireless sensor networks," *Mathematical and Computer Modelling*, vol. 54, no. 1-2, pp. 330–343, 2011.
- [2] H. H. Soliman, N. A. Hikal, and N. A. Sakr, "A comparative performance evaluation of intrusion detection techniques for hierarchical wireless sensor networks," *Egyptian Informatics Journal*, vol. 13, no. 3, pp. 225–238, 2012.
- [3] K. Shao, "A novel method of transformer fault diagnosis based on extension theory and information fusion in wireless sensor networks," *Energy Procedia*, vol. 12, pp. 669–678, 2011.
- [4] A. Díaz-Ramírez, L. A. Tafoya, J. A. Atempa, and P. Mejía-Alvarez, "Wireless sensor networks and fusion information methods for forest fire detection," *Procedia Technology*, vol. 3, pp. 69–79, 2012.
- [5] G. Li, J. He, and Y. Fu, "Group-based intrusion detection system in wireless sensor networks," *Computer Communications*, vol. 31, no. 18, pp. 4324–4332, 2008.
- [6] F. E. Asimakopoulou, G. J. Tsekouras, I. F. Gonos, and A. Ioannis, "Stathopoulos, estimation of seasonal variation of ground resistance using artificial neural networks," *Electric Power Systems Research*, vol. 94, pp. 113–121, 2013.
- [7] C. Oz and M. C. Leu, "American sign language word recognition with a sensory glove using artificial neural networks," *Engineering Applications of Artificial Intelligence*, vol. 24, no. 7, pp. 1204–1213, 2011.
- [8] I. Zaier, C. Shu, T. B. M. J. Ouarda, O. Seidou, and F. Chebana, "Estimation of ice thickness on lakes using artificial neural network ensembles," *Journal of Hydrology*, vol. 383, no. 3–4, pp. 330–340, 2010.
- [9] Q. Liu, X. Cui, M. F. Abbod, S. J. Huang, Y. Y. Han, and J. S. Shieh, "Brain death prediction based on ensembled artificial neural networks in neurosurgical intensive care unit," *Journal of the Taiwan Institute of Chemical Engineers*, vol. 42, no. 1, pp. 97–107, 2011.
- [10] S. Khoshjavan, B. Rezai, and M. Heidary, "Evaluation of effect of coal chemical properties on coal swelling index using artificial neural networks," *Expert Systems with Applications*, vol. 38, no. 10, pp. 12906–12912, 2011.
- [11] J. Yang, H. Singh, E. L. Hines et al., "Channel selection and classification of electroencephalogram signals: an artificial neural network and genetic algorithm-based approach," *Artificial Intelligence in Medicine*, vol. 55, no. 2, pp. 117–126, 2012.
- [12] P. Banerjee, V. S. Singh, K. Chattopadhyay, P. C. Chandra, and B. Singh, "Artificial neural network model as a potential alternative for groundwater salinity forecasting," *Journal of Hydrology*, vol. 398, no. 3–4, pp. 212–220, 2011.
- [13] C. H. Wu, Y. Ken, and T. Huang, "Patent classification system using a new hybrid genetic algorithm support vector machine," *Applied Soft Computing Journal*, vol. 10, no. 4, pp. 1164–1177, 2010.
- [14] J. Luts, G. Molenberghs, G. Verbeke, S. Van Huffel, and J. A. K. Suykens, "A mixed effects least squares support vector machine model for classification of longitudinal data," *Computational Statistics & Data Analysis*, vol. 56, no. 3, pp. 611–628, 2012.
- [15] D. C. Li and C. W. Liu, "A class possibility based kernel to increase classification accuracy for small data sets using support vector machines," *Expert Systems with Applications*, vol. 37, no. 4, pp. 3104–3110, 2010.
- [16] U. B. Parikh, B. Das, and R. Maheshwari, "Fault classification technique for series compensated transmission line using support vector machine," *International Journal of Electrical Power and Energy Systems*, vol. 32, no. 6, pp. 629–636, 2010.
- [17] H. Cevikalp, "New clustering algorithms for the support vector machine based hierarchical classification," *Pattern Recognition Letters*, vol. 31, no. 11, pp. 1285–1291, 2010.
- [18] D. Naiyang and T. Ying-Jie, *A New Data Mining Method—Support Vector Machine*, Science Press, Beijing, China, 2004.
- [19] T. Navalertporn and N. V. Afzulpurkar, "Optimization of tile manufacturing process using particle swarm optimization," *Swarm and Evolutionary Computation*, vol. 1, no. 2, pp. 97–109, 2011.
- [20] S. N. Deepa and G. Sugumaran, "Model order formulation of a multivariable discrete system using a modified particle swarm optimization approach," *Swarm and Evolutionary Computation*, vol. 1, no. 4, pp. 204–212, 2011.
- [21] H. Wang, H. Sun, C. Li, S. Rahnamayan, and J.-S. Pan, "Diversity enhanced particle swarm optimization with neighborhood search," *Information Sciences*, vol. 223, pp. 119–135, 2013.

## Research Article

# Solving Two-Dimensional HP Model by Firefly Algorithm and Simplified Energy Function

Yudong Zhang,<sup>1</sup> Lenan Wu,<sup>1</sup> and Shuihua Wang<sup>2</sup>

<sup>1</sup> School of Information Science and Engineering, Southeast University, Nanjing, Jiangsu 210096, China

<sup>2</sup> Grover School of Engineering, The City College of New York, New York, NY 10031, USA

Correspondence should be addressed to Lenan Wu; [wuln@seu.edu.cn](mailto:wuln@seu.edu.cn)

Received 18 December 2012; Accepted 9 January 2013

Academic Editor: Saeed Balochian

Copyright © 2013 Yudong Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In order to solve the HP model of the protein folding problem, we investigated traditional energy function and pointed out that its discrete property cannot give direction of the next step to the searching point, causing a challenge to optimization algorithms. Therefore, we introduced the simplified energy function into a turn traditional discrete energy function to continuous one. The simplified energy function totals the distance between all pairs of hydrophobic amino acids. To optimize the simplified energy function, we introduced the latest swarm intelligence algorithm, the firefly algorithm (FA). FA is a hot nature-inspired technique and has been used for solving nonlinear multimodal optimization problems in dynamic environment. We also proposed the code scheme strategy to apply FA to the simplified HP model with the clash test strategy. The experiment took 14 sequences of different chain lengths from 18 to 100 as the dataset and compared the FA with standard genetic algorithm and immune genetic algorithm. Each algorithm ran 20 times. The averaged energy convergence results show that FA achieves the lowest values. It concludes that it is effective to solve 2D HP model by the firefly algorithm and the simplified energy function.

## 1. Introduction

Protein folding is the process by which a protein structure assumes its functional shape or conformation. It is the physical process by which a polypeptide folds into its characteristic and functional three-dimensional structure from random coil. Each protein exists as an unfolded polypeptide or random coil when translated from a sequence of mRNA to a linear chain of amino acids [1]. This polypeptide lacks any developed three-dimensional structure. Amino acids interact with each other to produce a well-defined three-dimensional structure, the folded protein, known as the native state. The resulting three-dimensional structure is determined by the amino acid sequence [2, 3].

The protein folding has a challenging search space, since nature identifies the global minimum from more than  $10^{50}$  possible conformations for the backbone of a small protein [4]. A successful prediction requires two major components: (1) a set of free energy components for the protein, which are computationally inexpensive enough to be used in the search

procedure and sufficiently accurate to ensure the uniqueness of the native fold; (2) an efficient optimization procedure which is capable of finding an appropriate minimum for the strongly anisotropic function of hundreds of variables [5, 6].

Scholars tend to use the 2D lattice model (HP model) for protein folding. The HP model was proposed by Lau and Dill [7]. In this model, proteins consist of two different kinds of residues, hydrophobic and hydrophilic. The task is to minimize the energy function, which is defined as the counting of every two hydrophobic residues that are nonconsecutive nearest neighbors on the lattice [8].

The recent literatures solving the 2D HP model were reported as follows. Lin and Hsieh [9] proposed an efficient hybrid Taguchi genetic algorithm that combines genetic algorithm, Taguchi method, and particle swarm optimization, in order to enhance the performance of predicting protein structure. In addition, Lin presented the PSO inspired by a mutation mechanism in a genetic algorithm. In the experiment, Lin demonstrated that their algorithm can be applied successfully to the protein folding problems based on

the hydrophobic-hydrophilic lattice model. Zhang and Wu [10] investigated the bacterial chemotaxis optimization (BCO) on the 2D lattice model. He compared BCO with standard genetic algorithm, immune genetic algorithm, and artificial immune system for various chain lengths. He concluded that the BCO has the highest successful rate. Albrecht et al. [11] presented results from three-dimensional protein folding simulations in the HP model on ten benchmark problems. Their simulations are executed by a simulated annealing-based algorithm with a time-dependent cooling schedule. The neighborhood relation is determined by the pull-move set. The results of 10 benchmark problems provided experimental evidence of the relationship between the maximum depth, the stopping criterion, the mean of nonzero difference of the objective function, and their bounds. Chen et al. [12] suggested that protein folding cores form early in the process of folding, and proteins may have evolved to optimize both folding speed and native-state stability. Chen developed a set of empirical potential functions and applied them to analyze interaction energies among secondary structure elements in two proteins. Their work demonstrated that the predicted folding core also harbors residue that form native like interactions early in the folding reaction. Using a set of 29 unrelated proteins, Chen demonstrated that the average prediction result from their method is significantly better than predictions based on other computation methods. Sharma et al. [13] provided a description which is consistent with other natural processes, that the protein folding is formulated from the principle of increasing entropy. It then became evident that protein folding is an evolutionary process among many others. During the course of folding protein, structural hierarchy builds up in succession by diminishing energy density gradients in the quest for a stationary state determined by surrounding density in energy. Evolution toward more probable states, eventually attaining the stationary state, naturally selects steeply ascending paths on the entropy landscape that correspond to steeply descending paths on the free energy landscape. The dissipative motion of the non-Euclidian manifold is nondeterministic by its nature which clarifies why it is so difficult to predict protein folding. Zhang et al. [14] proposed a novel chaotic clonal genetic algorithm (CCGA) on a 2D lattice model. The algorithm combines chaos operator, clonal selection algorithm, and genetic algorithm. The experiments compared CCGA with standard genetic algorithm and immune genetic algorithm for various chain lengths, and found that CCGA not only find global minima more reliably, but also be significantly faster in convergence. Zhao [15] had introduced several natural computing approaches. In his paper, the evolutionary algorithm, memetic algorithm, ant colony optimization algorithm, tabu search, self-organizing map-based iterative approaches, and the typical chain growth computing approaches were reviewed. The advantages and disadvantages of various computing approaches and the current optimal results are analyzed.

However, those above mentioned algorithms cannot converge to the global optimal points within limited time. The reason lies at two sides. The first is that the energy function is discrete, so it cannot effectively guide the searching point

as the continuous energy function. The second is that the optimization algorithms themselves are not rapid enough.

To solve the two outstanding issues, we introduced a simplified energy function [16], and we introduced the latest firefly algorithm (FA). FA is a hot nature-inspired technique and has been used for solving nonlinear multimodal optimization problems in dynamic environment [17]. The algorithm is based on the behavior of the fireflies. In social insect colonies, each firefly seems to have its own plans, and yet the group acts as a whole appears to be highly organized. Scholars published immense literatures reporting that its performance, effectiveness, and robustness are superior to GA, PSO, and other global algorithms in a wide range of fields [17–19].

The structure of this paper was organized as follows. Section 2 gave a brief overview on the simplified 2D HP model and its energy function. Section 3 introduced the fundamental of firefly algorithm, giving its pseudocodes and the encoding schemes to simplified 2D HP model. Section 4 contained the experiment of 14 chains varying from 18 to 100 residues and compared the FA algorithm to standard genetic algorithm and immune genetic algorithm. Finally, Section 5 concluded the paper and summarized our contribution.

## 2. Overview of 2D HP Model

Proteins are the basis of biology. They are the driving force behind all of the biochemical reactions. They are the main constituent of our bones, muscles, hair, skin, and blood vessels. They recognize invading elements and allow the immune system to get rid of the unwanted invaders. For these reasons, scientists have sequenced the human genome, the blueprint for all of the proteins in biology. However, only knowing this sequence tells us little about what the protein does and how it does it. In order to carry out their function, they must take on a particular shape (fold). Therefore, proteins are genuinely amazing machines: before they do their work, they assemble themselves, which is called “folding.” [20] The protein folding problem is so difficult, so scholars prefer to use the 2D HP model proposed by Lau and Dill [7]. The task is to find the optimal folded configuration of a given sequence. Each potential configuration is associated with an energy function. The optimal configuration corresponds to the lowest energy.

*2.1. The 2D HP Model.* There are two types of amino acids in the realistic world: hydrophobic amino acid and hydrophilic amino acid. Since proteins always exist in the watery environment, the hydrophobic amino acid and hydrophilic amino acid tend to cling to each other, so as to apart from water to the maximal degree [16].

Figure 1 shows an 8-residue chain (0111110) before and after folding. We use 1 to denote hydrophobic residue (denoted by *red* circle), and 0 to denote hydrophilic residue (denoted by *green* circle). Figure 1 indicates that (1) the chain can turn 90° left or right, or continue ahead at each point; (2) the hydrophobic residues compact themselves to keep away from water.

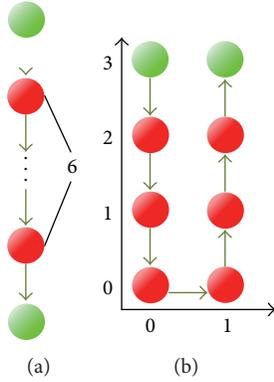


FIGURE 1: Illustration of HP model of protein chain 01111110: (a) before Folding; (b) after Folding.

**2.2. Simplified Energy Function.** In traditional HP model, the energy function is simple as adding  $-1$  for each direct (occupying nondiagonal neighboring lattice points) between two hydrophobic amino acids, those are not consecutive in the protein sequences. Figure 2(a) shows a protein sequence with energy function as  $-2$ .

However, it is not easy to pinpoint the direct and not consecutive two hydrophobic amino acids. Therefore, a simplified energy scoring method via distance matrix is proposed at Mathworks contest in 2002 [16]. We introduced in this strategy in our work. The energy is scored by totaling the distance between all pairs of hydrophobic amino acids. For Figure 2(b), we used the blue lines to link all pairs of hydrophobic amino acids, and summed distances is calculated as 21.129. The detailed calculation procedure is shown as follows.

*Step 1.* Chain = [0 1 1 1 1 1 1 0].

*Step 2.* Remove the hydrophilic residues (1st and last residues), and reserve the hydrophobic residues (2nd, 3rd, 4th, 5th, 6th, and 7th residues).

*Step 3.* Record their coordination as (0, 2), (0, 1), (0, 0), (1, 0), (1, 1), and (1, 2). See Figure 1(b).

*Step 4.* Calculate the residue-to-residue distances matrix as

$$\text{dist} = \begin{bmatrix} 0 & 1 & 2 & \sqrt{5} & \sqrt{2} & 1 \\ 1 & 0 & 1 & \sqrt{2} & 1 & \sqrt{2} \\ 2 & 1 & 0 & 1 & \sqrt{2} & \sqrt{5} \\ \sqrt{5} & \sqrt{2} & 1 & 0 & 1 & 2 \\ \sqrt{2} & 1 & \sqrt{2} & 1 & 0 & 1 \\ 1 & \sqrt{2} & \sqrt{5} & 2 & 1 & 0 \end{bmatrix}. \quad (1)$$

*Step 5.* Sum up the triangle part of the distance matrix

$$\text{Energy} = 7 + 4\sqrt{2} + 2\sqrt{5} + 2 * 2 = 21.129. \quad (2)$$

**2.3. Theory Analysis of Simplified Energy Function.** We expect that the simplified energy function is better than the traditional function. The reason is that the traditional energy

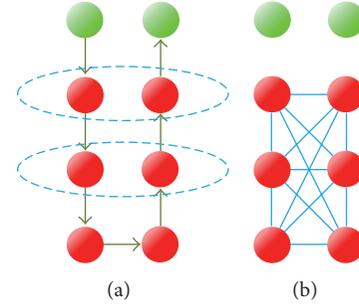


FIGURE 2: Comparison of two energy functions: (a) traditional energy function with energy as  $-2$ ; (b) simplified energy function with energy as 21.129.

function is discrete or stair wisely with  $-1$  decrease. It will not give instructive information for the searching point. While the simplified energy function used the sum of the distance matrix so as to transform traditional discrete energy function to a continuous one.

Figure 3 gave an illustration. Figure 3(a) shows the discrete energy function, where the searching point found the energy function remains as 1 within the searching area, so it can not give the direction of the next step for the searching point. Figure 3(b) shows the continuous energy function, where the searching point found the energy function varies, so it will move towards the minimal point within the searching area.

### 3. Firefly Algorithm

A power optimization algorithm is of importance to solve 2D HP model. Firefly Algorithm (FA) is a nature-inspired algorithm based on the flashing behaviors of the firefly swarm [19]. The primary purpose for the flash of fireflies is to signal to attract other fireflies. The assumption of FA consists of three rules: (1) all fireflies are unisex, so that one firefly will be attracted to other fireflies regardless of their sex; (2) an important and interesting behavior of fireflies is to glow brighter mainly to attract prey and to share food with others; (3) attractiveness is proportional to their brightness, thus each agent firstly moves toward a neighbor that glows brighter.

In the FA, the fireflies are randomly distributed in the search space. The fireflies carry a luminescence quality, called luciferin, which emits light proportional to the quality [18]. Each firefly is attracted by the brighter glow of other approximated fireflies. The attractiveness decreases as their distance increases [21]. If there is no brighter one within the scope of a firefly, it will move randomly in the search space.

**3.1. Variation of Light Intensity.** The brightness is related to the objective values, so for an optimization problem, a firefly with higher intensity will attract another firefly with higher probability, and vice versa. Assume that there exists a swarm of  $n$  fireflies, and  $x_i$  represents a solution for a firefly  $i$ , whereas  $f(x_i)$  denotes its corresponding energy value. Here,

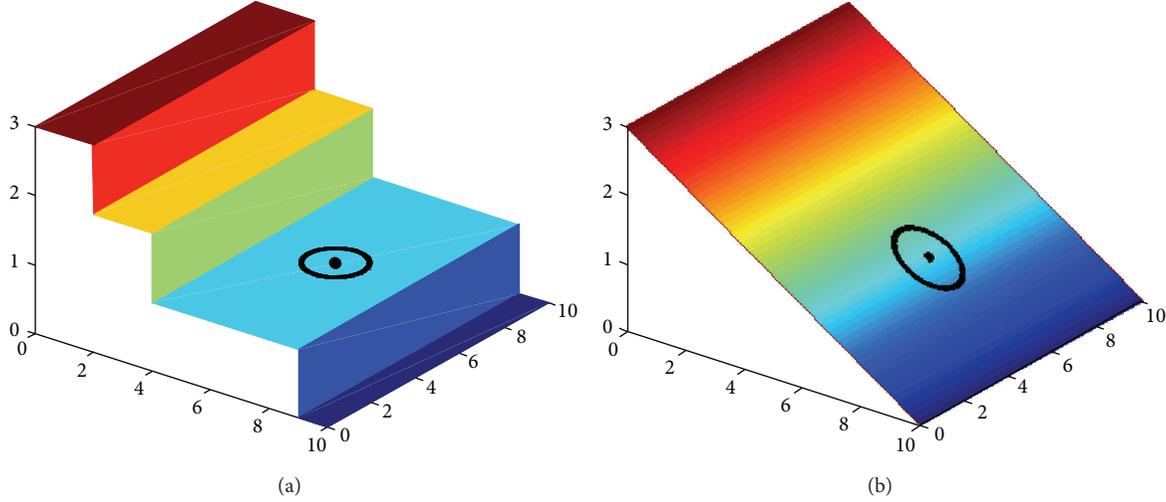


FIGURE 3: The effect of energy function on the optimization procedures: (a) discrete energy function; (b) continuous energy function. The black dot denotes the searching point, and the black circle denotes the searching territory.

the brightness  $I$  of a firefly is equivalent to the simplified energy value

$$I_i = f(x_i), \quad 1 \leq i \leq n. \quad (3)$$

**3.2. Movement towards Attractive Flies.** The attractiveness  $\beta$  of the firefly is proportional to the light intensity received by the adjacent fireflies [22]. Suppose  $\beta_0$  is the attractiveness with distance  $r = 0$ , so for two fireflies  $i$  and  $j$  at locations  $x_i$  and  $x_j$ , their attractiveness is calculated as

$$\beta_r(i, j) = \beta_0 \exp\{-\gamma r(i, j)^2\} \quad (4)$$

$$r(i, j) = \|x_i - x_j\|,$$

where  $r(i, j)$  denotes the distance between fireflies  $i$  and  $j$ ,  $\gamma$  denotes the light absorption coefficient. Suppose that firefly  $j$  is brighter than firefly  $i$ , then firefly  $i$  will move to a new location as

$$x_i(t+1) = x_i(t) + \beta_0 \exp\{-\gamma r^2\}(x_j - x_i). \quad (5)$$

**3.3. Pseudocodes.** See Pseudocode 1.

**3.4. Encoding Scheme.** For the 2D Hp model, we set the firefly as a string of digits chosen from the set  $\{0, 1, 2\}$ . Each digit denotes a torsion angle as:  $0 \rightarrow$  left,  $1 \rightarrow$  right, and  $2 \rightarrow$  continue [8]. The initialization of FA generated the fireflies with random digits, and the first link of the sequence is not included in the encoding. Figure 4 shows that the direction of the first link only makes the structure rotated. Their structures of the lattice model does not change, neither does the corresponding energy levels.

**3.5. Clash Test.** Sometimes the decoded string represents a clash, which means a lattice node is occupied by more than

one residue, as shown in Figure 5. If a clash occurred, this firefly will be replaced by a new randomly generated firefly. If the clash occurs in the new one, then the procedures are repeated until no clash occurs. Meanwhile, the clash tests need to be checked at every step after the fireflies are updated, see (5) for detailed information.

## 4. Experiments

The experiments were carried out on the platform of P4 IBM with 3.1 GHz processor and 2 G RAM, running under Windows XP operating system. The algorithm was in-house developed via Matlab 2012a. The experiment data contains 14 protein folding problems with different chain lengths varying from 18 to 100 generated randomly. Their sequences are listed in Table 1.

**4.1. Effectiveness of Simplified Energy Function.** Figure 6 gave the optimal solutions obtained by FA through the simplified HP model. It forces the hydrophobic residues to fold themselves together to the utmost extent. Obviously, all the hydrophobic residues are located at the inner side, while the hydrophilic residues are located outside. It satisfies the expectation of 2D HP model. Therefore, this simplified energy function is effective.

**4.2. Algorithm Comparison.** We compared the FA with traditional standard genetic algorithm (SGA) and immune genetic algorithm (IGA). We ran each algorithm 20 times and got the averaged converged energy values of total 20 times results. The parameters are set by hundreds of experiments to get the optimal values and are shown in Table 2.

The average convergence results were illustrated in Figure 7. It indicated that the firefly algorithm achieves the least averaged convergence values among all three algorithms

```

The pseudo-codes of the firefly algorithm can be summarized as
Step 1 Initialization.
    Step 1.1 Create the initial population of  $n$  fireflies ( $x_1, x_2, \dots, x_n$ ) within  $d$ -dimensional search space.
    Step 1.2 Formulate light intensity of each firefly so as to be associated with the energy value  $f(x)$ .
    Step 1.3 Define the parameters  $\beta_0$  and  $\gamma$ .
Step 2 Perform.
while (termination criteria are not met)
    for  $i = 1$  to  $n$ 
        for  $j = 1$  to  $n$ 
            if ( $I_j < I_i$ )
                Move firefly  $j$  towards firefly  $i$  via (5).
            end if
        Update Attractiveness Table.
        Evaluate New Solutions and Update light intensity.
    end for  $j$ 
end for  $i$ 
Rank the fireflies and find the best.
end for while
Step 3 Post-processing and output.
    
```

PSEUDOCODE 1

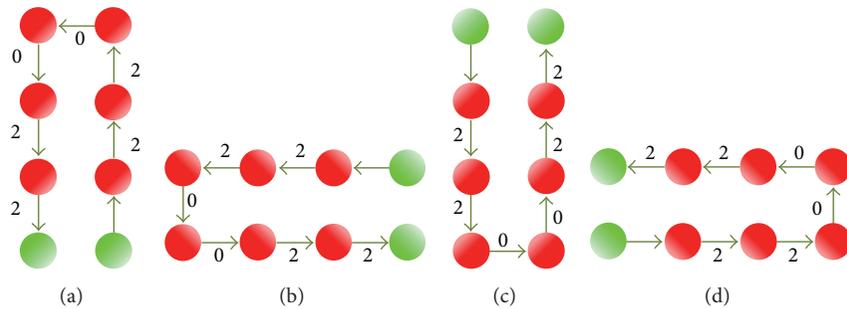


FIGURE 4: The direction of the first link does not contribute to the encoding. The first link directs to: (a) North, (b) West, (c) South, and (d) East does not affect the protein structure. They are all encoded as “220022”, and their energies are all 21.129.

for all protein sequences. Therefore, the FA is the most effective among the three algorithms. Besides, the longer the proteins become (the index increases), the larger the advantages of FA turn.

**4.3. Energy Function Comparison.** In this chapter, we compared the simplified energy function with traditional energy function. We choose the first protein in the dataset, of which the minimal traditional energy function is  $-5$ , and the minimal simplified energy function is  $100.97$ . Their convergence curves are shown in Figure 8.

We see that the simplified energy function merely costs about 1/3 iterations of those needed by the traditional energy function. The reason lies in the fact that the traditional energy function is discrete and cannot give effective direction for the searching point to move to, whereas the simplified energy function can be regarded as a continuous function, so the searching point is easy to guide itself moving towards the minimal point within its search territory. Moreover, Figure 8 validates that our developed theory in Section 2.3 is coherent to the experimental results.

### 5. Conclusions and Future Work

This study introduced the firefly algorithm and the simplified energy function and applied them to the 2D HP lattice model of protein folding problem. The experimental results on 20 runs of 14 chains with different lengths showed that the FA can achieve the lowest averaged energy values compared to standard genetic algorithm and immune genetic algorithm. We also prove the superiority of the simplified energy function on traditional energy function.

Our contributions can be summarized as follows.

- (1) We applied firefly algorithm to the field of protein folding problem.
- (2) We introduced the simplified energy function.
- (3) We demonstrate the superiority of the simplified energy function on traditional energy function on both from the discrete and continuous energy function theory and experimental results.

The future work focuses on three points. First is to investigate more novel intelligence algorithms, including

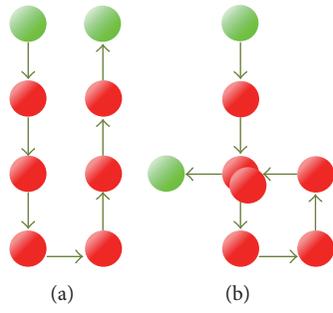


FIGURE 5: Illustration of clashes: (a) no Clash (encoded as “220022”); (b) clash (encoded as “220002”).

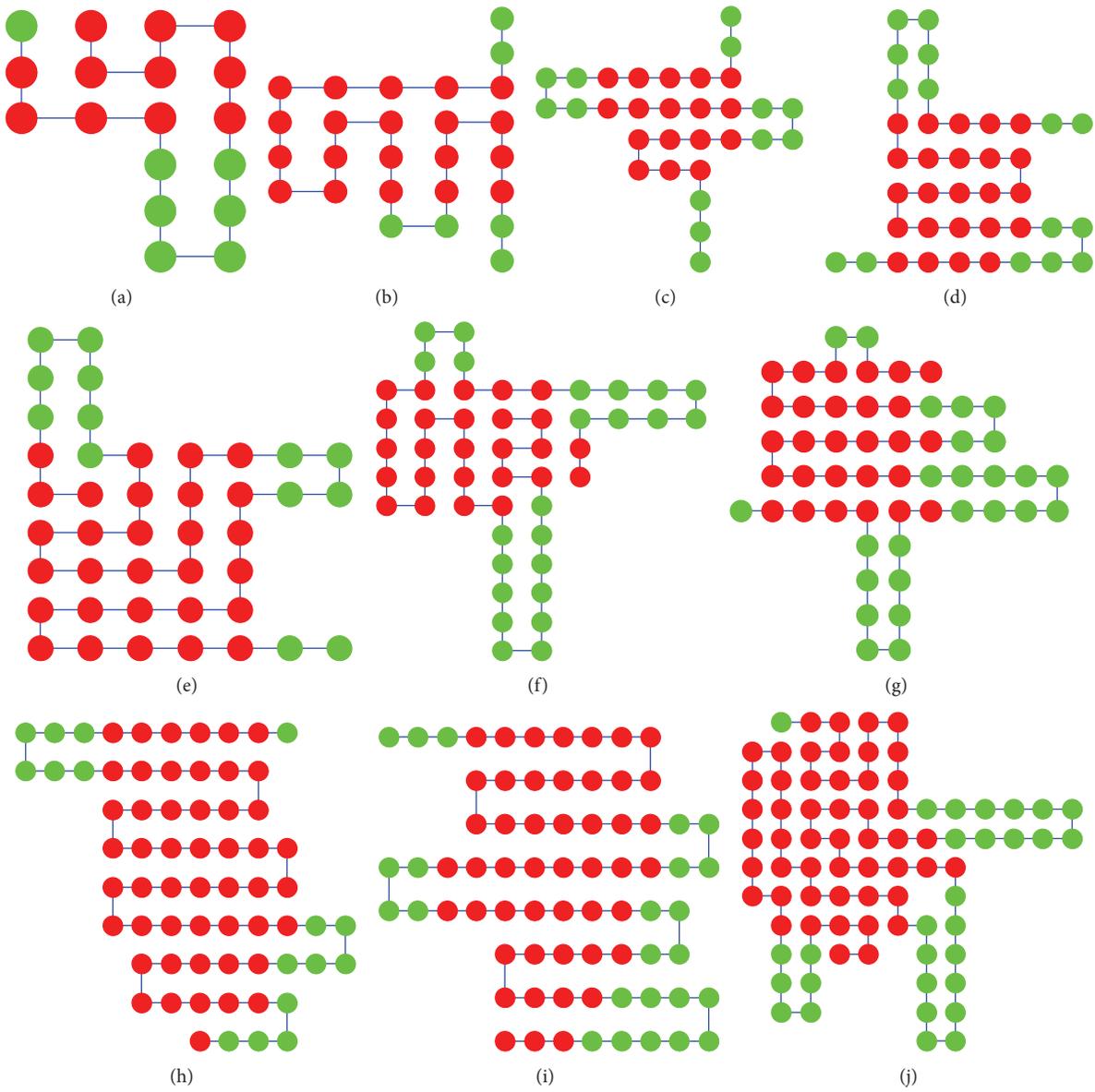


FIGURE 6: Continued.

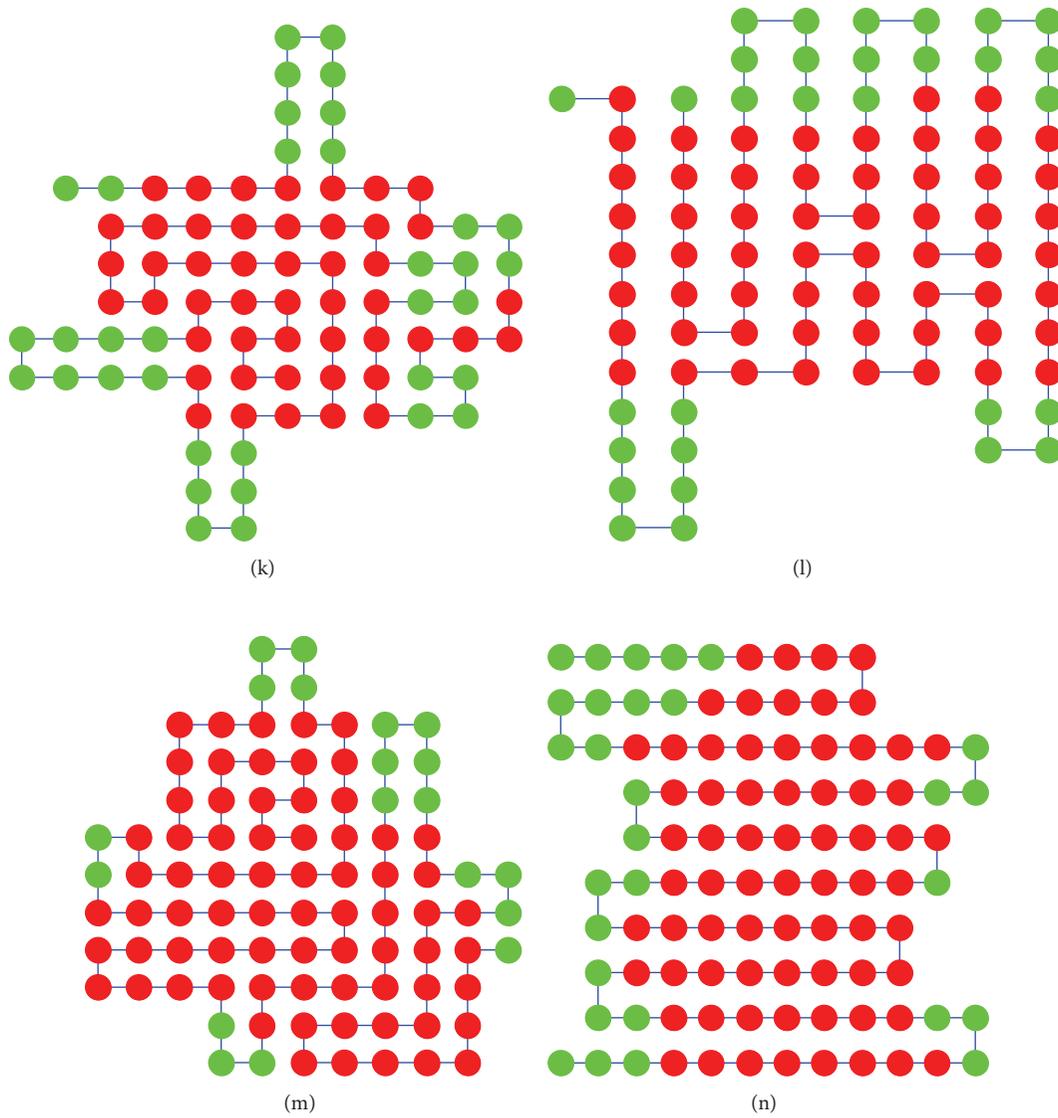


FIGURE 6: Optimal solutions of the 14 sequences found by FA: (a) 18 residues; (b) 26 residues; (c) 30 residues; (d) 39 residues; (e) 42 residues; (f) 49 residues; (g) 53 residues; (h) 66 residues; (i) 72 residues; (j) 79 residues; (k) 83 residues; (l) 89 residues; (m) 91 residues; (n) 100 residues.

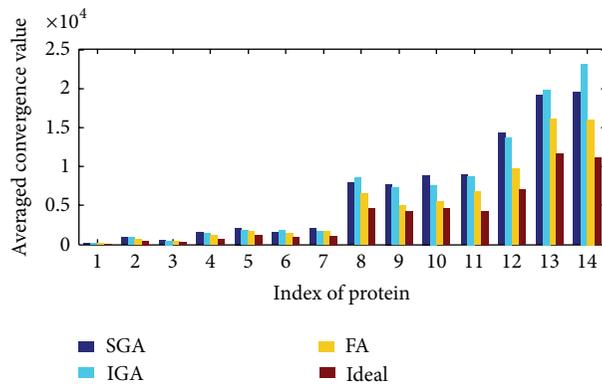


FIGURE 7: The algorithm comparison by averaged convergence values.

TABLE 1: 14 protein chains of different residues.

Index	Length	Energy	Protein Sequence
1	18	100.97	011110000001111111
2	26	457.23	00111111001111111111111100
3	30	305.16	000111111100001111100001111100
4	39	716.77	0011110000011111111111111000000111100
5	42	1174.79	0011111111111100001111111111110000000111
6	49	885.69	1111110000000000011111111111100001110000000011
7	53	1082.36	011110000000011000000000111111111100000111111100111
8	66	4587.09	1000011111111100000111111111111111111111111110000001111110
9	72	4168.48	1110000000001111111100001111110000111111100001111111111111111111111111 1000
10	79	4611.02	0111111111111100000000000111111111100000000000111111111111111111111111 00000011111
11	83	4319.72	00111100000000011110001111000011110000111111111111111111111100000011 0000000011111111
12	89	6981.92	0111111100000000011111111111110000111111000001111111111000001111 11000000111111111110
13	91	11721.87	01111111111111110001100000011111111000111111111111111110011111111111 11000011111111111111
14	100	11175.25	00011111111000111111100011111111111110001111110111111100111111100 0111111110000001111111100000

TABLE 2: Parameters setting.

Algorithm	Parameters
SGA	Population ( $N$ ) = 100, cross rate ( $P_C$ ) = 0.3, mutation rate ( $P_M$ ) = 0.1
IGA	Population ( $N$ ) = 100, cross rate ( $P_C$ ) = 0.3, mutation rate ( $P_M$ ) = 0.1, elimination ratio ( $r$ ) = 0.1
FA	Population ( $N$ ) = 100, attractiveness with 0 distance ( $\beta_0$ ) = 1, light absorption coefficient ( $\gamma$ ) = 1

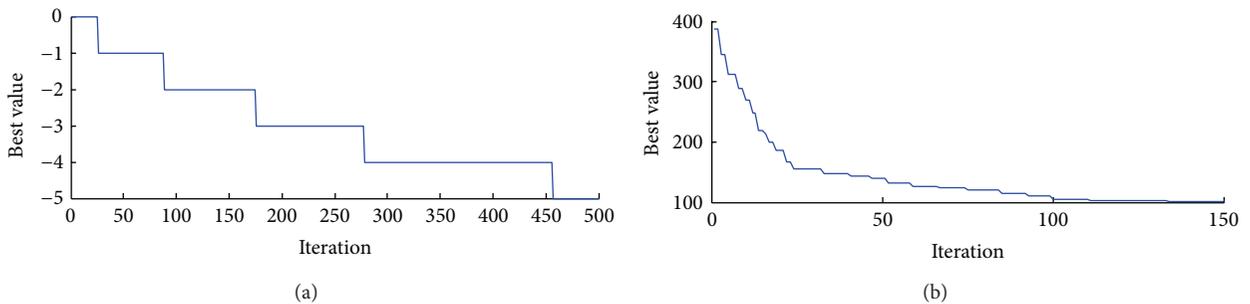


FIGURE 8: Energy function comparison: (a) traditional energy function; (b) simplified energy function.

differential algorithm and intelligent water. We expect that they will give excellent performances after being adapted to the field of HP model; second is to analyze the principal of FA and improve its performance; third is to analyze the mathematical fundamental of simplified energy function and try to give more logical proof of its superiority.

References

[1] S. K. Sharma, P. Goloubinoff, and P. Christen, "Heavy metal ions are potent inhibitors of protein folding," *Biochemical and*

*Biophysical Research Communications*, vol. 372, no. 2, pp. 341–345, 2008.  
 [2] B. Turner and D. M. J. Lilley, "The importance of G-A hydrogen bonding in the metal ion- and protein-induced folding of a Kink Turn RNA," *Journal of Molecular Biology*, vol. 381, no. 2, pp. 431–442, 2008.  
 [3] D. Tomkiewicz, N. Nouwen, and A. J. M. Driessen, "Kinetics and energetics of the translocation of maltose binding protein folding mutants," *Journal of Molecular Biology*, vol. 377, no. 1, pp. 83–90, 2008.  
 [4] O. Fedorova, A. Solem, and A. M. Pyle, "Protein-facilitated folding of group II intron ribozymes," *Journal of Molecular Biology*, vol. 397, no. 3, pp. 799–813, 2010.

- [5] B. C. Gin, J. P. Garrahan, and P. L. Geissler, "The limited role of nonnative contacts in the folding pathways of a lattice protein," *Journal of Molecular Biology*, vol. 392, no. 5, pp. 1303–1314, 2009.
- [6] Y. Zhang, S. Wang, L. Wu, and Y. Huo, "Artificial immune system for protein folding model," *Journal of Convergence Information Technology*, vol. 6, no. 1, pp. 55–61, 2011.
- [7] K. F. Lau and K. A. Dill, "Lattice statistical mechanics model of the conformational and sequence spaces of proteins," *Macromolecules*, vol. 22, no. 10, pp. 3986–3997, 1989.
- [8] R. König and T. Dandekar, "Improving genetic algorithms for protein folding simulations by systematic crossover," *BioSystems*, vol. 50, no. 1, pp. 17–25, 1999.
- [9] C. J. Lin and M. H. Hsieh, "An efficient hybrid Taguchi-genetic algorithm for protein folding simulation," *Expert Systems with Applications*, vol. 36, no. 10, pp. 12446–12453, 2009.
- [10] Y. Zhang and L. Wu, "Bacterial chemotaxis optimization for protein folding model," in *Proceedings of the 5th International Conference on Natural Computation (ICNC '09)*, pp. 159–162, Tianjian, China, August 2009.
- [11] A. A. Albrecht, A. Skaliotis, and K. Steinhöfel, "Stochastic protein folding simulation in the three-dimensional HP-model," *Computational Biology and Chemistry*, vol. 32, no. 4, pp. 248–255, 2008.
- [12] M. Chen, A. D. Dousis, Y. Wu, P. Wittung-Stafshede, and J. Ma, "Predicting protein folding cores by empirical potential functions," *Archives of Biochemistry and Biophysics*, vol. 483, no. 1, pp. 16–22, 2009.
- [13] V. Sharma, V. R. I. Kaila, and A. Annala, "Protein folding as an evolutionary process," *Physica A*, vol. 388, no. 6, pp. 851–862, 2009.
- [14] Y. Zhang, L. Wu, Y. Huo, and S. Wang, "Chaotic clonal genetic algorithm for protein folding model," in *Proceedings of the International Conference on Computer Application and System Modeling (ICCASM '10)*, pp. V3120–V3124, October 2010.
- [15] X. Zhao, "Advances on protein folding simulations based on the lattice HP models with natural computing," *Applied Soft Computing Journal*, vol. 8, no. 2, pp. 1029–1040, 2008.
- [16] "Protein Folding Contest Rules," <http://www.mathworks.com/matlabcentral/contest/contests/11/rules>.
- [17] J. Senthilnath, S. N. Omkar, and V. Mani, "Clustering using firefly algorithm: performance study," *Swarm and Evolutionary Computation*, vol. 1, no. 3, pp. 164–171, 2011.
- [18] M.-H. Horng and R.-J. Liou, "Multilevel minimum cross entropy threshold selection based on the firefly algorithm," *Expert Systems With Applications*, vol. 38, no. 12, pp. 14805–14811, 2011.
- [19] X.-S. Yang, S. S. S. Hosseini, and A. H. Gandomi, "Firefly Algorithm for solving non-convex economic dispatch problems with valve loading effect," *Applied Soft Computing*, vol. 12, no. 3, pp. 1180–1186, 2012.
- [20] "The science behind Folding," <http://folding.stanford.edu/English/Science>.
- [21] A. H. Gandomi, X.-S. Yang, and A. H. Alavi, "Mixed variable structural optimization using Firefly Algorithm," *Computers & Structures*, vol. 89, no. 23–24, pp. 2325–2336, 2011.
- [22] M.-H. Horng, "Vector quantization using the firefly algorithm for image compression," *Expert Systems With Applications*, vol. 39, no. 1, pp. 1078–1091, 2012.